Review

Self-supervised anomaly detection in computer vision and beyond: A survey and outlook

Hadi Hojjati [*,1], Thi Kieu Khanh Ho[1], Narges Armanfard

Department of Electrical and Computer Engineering, McGill University, Montreal, QC, Canada
Mila - Quebec AI Institute, Montreal, QC, Canada

ARTICLE INFO

Keywords:
Anomaly detection
Self-supervised learning
Contrastive learning
Representation learning

ABSTRACT

Anomaly detection (AD) plays a crucial role in various domains, including cybersecurity, finance, and healthcare, by identifying patterns or events that deviate from normal behavior. In recent years, significant progress has been made in this field due to the remarkable growth of deep learning models. Notably, the advent of self-supervised learning has sparked the development of novel AD algorithms that outperform the existing state-of-the-art approaches by a considerable margin. This paper aims to provide a comprehensive review of the current methodologies in self-supervised anomaly detection. We present technical details of the standard methods and discuss their strengths and drawbacks. We also compare the performance of these models against each other and other state-of-the-art anomaly detection models. Finally, the paper concludes with a discussion of future directions for self-supervised anomaly detection, including the development of more effective and efficient algorithms and the integration of these techniques with other related fields, such as multi-modal learning.

1. Introduction

Anomaly detection (AD) is the task of identifying samples that differ significantly from the majority of data and often signals an irregular, fake, rare, or fraudulent observation (Wang, Bah, & Hammad, 2019). Anomaly detection is particularly useful in cases where we cannot define all existing classes during training. This makes AD algorithms applicable to a broad range of applications, including but not limited to intrusion detection in cybersecurity (Xin et al., 2018), fraud detection (Malaiya et al., 2018), acoustic novelty detection (Hojjati & Armanfard, 2022), and medical diagnosis (Latif, Usman, Rana, & Qadir, 2018).

In the past, anomaly detection relied on manual inspection of data by experts. However, with the proliferation of sensory systems, the volume of data has surged, making the traditional method impractical. As a result, automatic anomaly detection methods, including machine learning (ML)-based techniques, have gained significant popularity. Over the past few decades, numerous ML-based models have been developed for this purpose. Classical approaches like Kernel Density Estimation (KDE), One-Class Support Vector Machine (OCSVM), and Isolation Forests (IF) have been widely adopted. However, the performance of these algorithms often degrades when applied to higher-dimensional data. In recent years, deep learning models have shown significant improvements over traditional ML models since they have the capability to learn intricate patterns and representations from vast amounts of data, making them well-suited for anomaly detection. The utilization of deep learning for anomaly detection has yielded high accuracy and robust results, establishing it as a popular choice in various applications (Hojjati & Armanfard, 2023; Ruff et al., 2021).

Deep-learning based models for anomaly detection can be broadly classified into three categories: The first category comprises models that utilize deep neural networks to learn a lower-dimensional representation of high-dimensional data. Subsequently, they apply a classical anomaly detection algorithm, such as One-Class Support Vector Machine (OCSVM) (Schölkopf, Williamson, Smola, Shawe-Taylor, & Platt, 1999), to the obtained lower-dimensional representation (Sabokrou, Fayyaz, Fathy, Moayed, & Klette, 2018). By mapping the data into a lower-dimensional space, these approaches mitigate the curse of dimensionality issues associated with traditional non-deep learning anomaly detection methods, thereby yielding reasonably accurate detection results. The second group of methods involves using deep neural networks to reconstruct the input data and calculate an anomaly score directly based on the data reconstruction loss. The prevalent network architectures employed in these methods are Autoencoders (AEs) (Chen, Yeo, Lee, & Lau, 2018) and Generative Adversarial Networks (GANs)

* Corresponding author at: Department of Electrical and Computer Engineering, McGill University, Montreal, QC, Canada.
E-mail address: hadi.hojjati@mcgill.ca (H. Hojjati).
[1] Both authors contributed equally to the paper.

(Schlegl, Seeböck, Waldstein, Langs, & Schmidt-Erfurth, 2019; Schlegl, Seeböck, Waldstein, Schmidt-Erfurth, & Langs, 2017). The underlying assumption is that a network trained solely on reconstructing normal data will produce a significant reconstruction error when confronted with an anomaly. The third category encompasses algorithms combining both approaches (Hojjati & Armanfard, 2023; Ruff et al., 2018). These methods jointly train a neural network for feature extraction and an anomaly detector on the latent space of the network. The anomaly detector assigns an anomaly score to input data based on the learned representations. By combining feature extraction and anomaly detection in a unified framework, these models aim to enhance detection performance. Although the above methods use different approaches for AD, the concept remains the same, i.e. normal samples have similar feature distribution in the latent space of the trained network, and abnormal instances are not in line with the ordinary anticipated behavior of normality.[2]

Compared to typical deep learning tasks, anomaly detection poses unique challenges due to the characteristics of the data involved. Anomalies are typically rare occurrences or costly events in the real world. Consequently, the training data for anomaly detection is imbalanced, with a majority of normal data and only a small number of anomalies. Moreover, these anomalous samples can be contaminated with noise, further complicating the detection task. Additionally, anomalies cannot be treated as a single class, and a detection system may encounter new types of abnormalities that were not present in the training data. These challenges render a significant portion of deep learning algorithms ineffective for anomaly detection.

In general, deep learning models can be categorized into supervised, semi-supervised, and unsupervised methods. Supervised methods, which rely on labeled data, often achieve high performance. However, as previously mentioned, annotated data is not commonly available for anomaly detection tasks, making semi-supervised and unsupervised models the only practical options. Unfortunately, these algorithms generally do not perform as well as their supervised counterparts. This limitation acts as a significant bottleneck, preventing deep anomaly detection algorithms from surpassing a certain performance threshold.

Recently, there has been a resurgence of hope for anomaly detection algorithms with the emergence of self-supervised learning (SSL). In SSL, similar to unsupervised learning, the model learns from unlabeled data without external annotation. It learns a generalizable representation from data by solving a supervised proxy task which is often unrelated to the target task but can help the network to learn a better embedding space. Depending on the nature of the data, a diverse set of tasks, such as colorization (Larsson, Maire, & Shakhnarovich, 2016), mutual information maximization (Hjelm et al., 2019), and predicting geometric transformations (Gidaris, Singh, & Komodakis, 2018) can be used as the supervised proxy task. These methods showed promising results in various applications, such as speech representation learning (Ravanelli et al., 2020), visual feature learning (Jing & Tian, 2021), and healthcare applications (Azizi et al., 2021). Even in some cases, self-supervised algorithms have approached the performance of fully-supervised models (Chen, Kornblith, Norouzi, & Hinton, 2020). Additionally, self-supervised models can learn representations that capture complex patterns and relationships in the data, making them effective at detecting subtle anomalies that other methods might miss.

Motivated by the recent success of SSL, anomaly detection researchers have started to incorporate the idea of self-supervision for developing effective algorithms. Their studies showed that the representation that is learned through self-supervision could be useful for anomaly detection if the anomaly score and the pretext task are defined appropriately (Reiss & Hoshen, 2021; Tack, Mo, Jeong, & Shin, 2020). As a result, self-supervised algorithms have emerged as the new state-of-the-art in anomaly detection, outperforming other traditional methods. Recently, a wide range of SSL frameworks has been developed for anomaly detection. However, to the best of our knowledge, no paper conducted a comprehensive review of these methods. We aim to fill this gap by thoroughly reviewing and categorizing self-supervised learning approaches in anomaly detection. Our work provides a valuable resource for researchers and practitioners in this field and contributes to advancing state-of-the-art anomaly detection. In short, we can summarize the contribution of our work as follows:

- In a pioneering contribution to the existing literature, we present a cohesive overview of self-supervised methods for anomaly detection for the first time. This distinctive approach unifies these methods irrespective of the data type they handle.
- We categorize existing self-supervised anomaly detection algorithms into two overarching groups based on their requirement for negative samples during training. Within each category, we further classify these algorithms based on their proxy tasks.
- We conduct a comprehensive performance comparison between self-supervised methods and traditional algorithms, thoroughly discussing their respective strengths and weaknesses.
- To conclude, we offer insights into potential future directions in self-supervised anomaly detection research, providing a roadmap for further exploration and advancement in this evolving field.

2. Related works

In recent decades, there has been significant research and exploration of the anomaly detection problem across various domains. Several survey articles attempted to group anomaly detection algorithms into distinctive categories. Hodge and Austin (2004) and Agyemang, Barker, and Alhajj (2006) are two examples of early studies that categorized the existing algorithms and extensively discussed the techniques that are used in each category. In another prominent work, Chandola, Banerjee, and Kumar (2009) surveyed the existing anomaly detection algorithms and divided them into distinctive categories. In addition to describing the technical details of each method, they identified the underlying assumptions that are implicitly made regarding the anomalies. They also discussed the advantages, disadvantages and computational complexity of each technique. Furthermore, they extensively reviewed the application areas of the methods and highlighted the challenges faced in each domain.

More recently, deep learning methods have inspired researchers in anomaly detection, leading to the development of new algorithms in this domain. As a result, review papers focusing on deep anomaly detection have emerged. Chalapathy and Chawla (2019) was one of the first papers that presented a comprehensive review of deep anomaly detection methods. They categorized the existing algorithms based on their underlying assumptions and explained the pros and cons of each approach. Chalapathy and Chawla (2019) have also thoroughly explored applications of deep anomaly detection and assessed the effectiveness of each method. In another similar survey, Pang, Shen, Cao, and Hengel (2021) reviewed contemporary deep AD methods. They first discussed the challenges and complexities that anomaly detection faces, and then they categorized the existing deep methods into three high-level categories and eleven fine-grained subcategories. They emphasized how each category addresses challenges and identified key assumptions and intuitions. Notably, they also compiled a list of publicly available codes and datasets for benchmarking. While most review papers in recent years focused on specific sets of algorithms, Ruff et al. (2021) presented an extensive survey of anomaly detection methods, unifying classic shallow methods with recent deep approaches. They highlighted connections and similarities between these two types of algorithms, providing an in-depth description and taxonomy of common practices and challenges in anomaly detection. In addition to the mentioned

[2] In the rest of the paper, the term *normal* has no relationship with the normal Gaussian distribution unless specified otherwise.

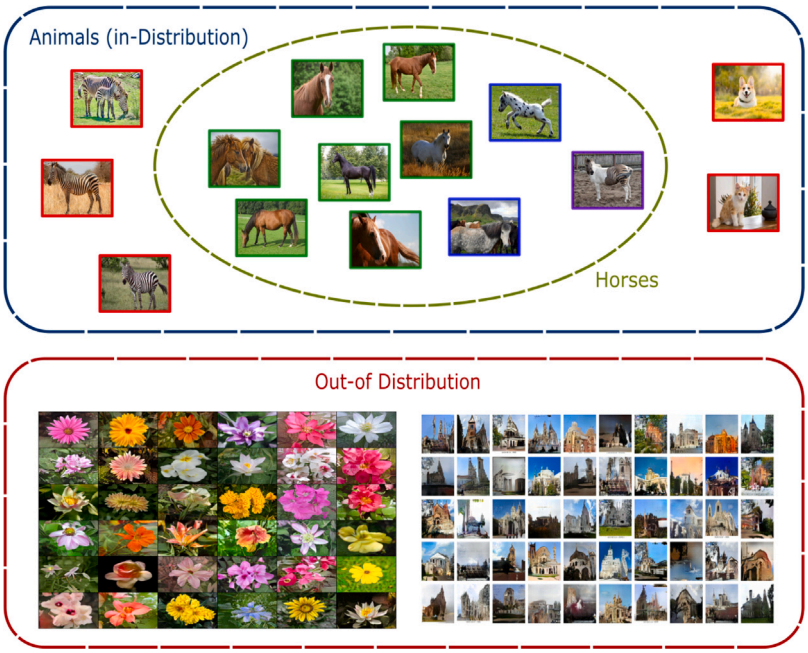


Fig. 1. Normal samples are shown in green, anomalies in red, outliers in blue and novelties in purple. The dataset of animals is denoted by a light-blue dashed box while a dashed dark-red box shows other out-of-distribution datasets.

studies, several other review papers in this field have been published, focusing on specific domains of application or particular types of methods. For example, the two survey papers Di Mattia, Galeone, De Simoni, and Ghelfi (2019) and Xia et al. (2022) are dedicated to reviewing the GAN-based anomaly detection methods. They discussed these models' theoretical bases and practical applications and provided a detailed description of existing challenges and future directions in GAN-based anomaly detection. Both of the papers also carried out empirical evaluations to compare the performance of different algorithms. In another study, Villa-Perez et al. (2021) empirically evaluated the performance of 29 semi-supervised AD algorithms.

While numerous survey papers have explored various aspects of anomaly detection, there remains a research gap concerning the thorough investigation of self-supervised methods, which have emerged as state-of-the-art in recent years. This paper aims to address this gap and provide a comprehensive analysis of self-supervised anomaly detection papers.

3. Anomaly detection: Terminology and common practices

The term *anomaly detection* is commonly used to encompass all algorithms designed to identify samples that deviate from normal patterns. Needless to say, the development of anomaly detection models depends on factors such as the availability of data labels, types of anomalies, and specific applications. Furthermore, there is inconsistency in the nomenclature used in the literature. To ensure clarity and avoid confusion, we first define and describe the relevant terminologies used throughout the paper.

3.1. Anomaly, outlier, novelty, out-of-distribution detection

Some studies use the terms *anomaly*, *novelty*, *outlier* and *out-of-distribution* interchangeably, while others distinguish them. Although most of the algorithms for detecting them are similar, their significance and application might differ. In this paper, we adopt the terminology proposed by previous studies and define each task as follows (Ruff et al., 2021):

- **Anomaly Detection:** *Anomaly detection* can be defined as the task of identifying samples that are drawn from a distribution other than the distribution of normal instances, denoted as $\mathbb{P}^+$. For instance, if we consider $\mathbb{P}^+$ as the distribution of horses, a zebra would be considered an anomaly in the context of anomaly detection.
- **Outlier Detection:** An *outlier* is defined as a low-probability sample drawn from the distribution of normal instances, $\mathbb{P}^+$. For instance, in the context of horse detection, a Falabella (a small horse breed) would be considered an outlier among the various horse breeds.
- **Novelty Detection** A *novelty* is a sample that is drawn from a new region of a non-stationary distribution of normal samples $\mathbb{P}^+$. These samples are often encountered during the inference phase, but their counterparts were not present in the training data. For instance, a new breed of horses is considered a novelty in the horse detection task.
- **Out-of-Distribution Detection** In *out-of-distribution (OOD)* detection, the goal is to identify samples that do not belong to any of the training set classes. This problem, which is also referred to as *open category detection* (Liu, Garrepalli, Dietterich, Fern, & Hendrycks, 2018), is often formulated as a supervised problem where we have the labeled data from K classes during training. We treat all the K classes as normal and aim to identify if a sample does not come from these classes during the inference phase. Recent studies have shown that training a supervised classifier on K classes and using Softmax probabilities for calculating the anomaly scores can yield state-of-the-art performance in the OOD detection task (Vaze, Han, Vedaldi, & Zisserman, 2021) An example of the OOD detection task is using a classifier trained on an animal dataset to detect samples from other datasets, e.g. flowers.

Fig. 1 illustrates an example of normal sample versus anomalies, outliers, novelty and out-of-distribution data.

3.2. Types of anomalies

In the classic anomaly detection literature, anomalies are classified into three categories based on their nature (Chandola et al., 2009; Pang et al., 2021):

- **Point Anomalies:** A *point anomaly* refers to an individual sample that exhibits an irregularity or deviation from the standard pattern. A single cat image in the dataset of dog images or a fraudulent insurance claim are examples of point anomalies. Most studies in the anomaly detection literature focus on this type of anomaly (Chalapathy & Chawla, 2019).
- **Contextual Anomalies:** A *contextual anomaly*, also known as a *conditional anomaly*, is a data point deemed abnormal within a specific context. The context should be defined as a part of the problem formulation. For instance, a value of 120 km/h is considered an abnormal recording of the speed of a bike, whereas it is not considered an abnormal recording of the speed of a car. The anomaly classification depends on the context in which the data point is evaluated.
- **Collective Anomalies:** *Collective anomalies*, also called *group anomalies*, are a subset of data points that exhibit collective abnormality when considered in relation to the entire dataset. While each sample within a collective anomaly may not be abnormal, their combined presence indicates an anomaly. For instance, a series of high-value credit card transactions that occur rapidly and consecutively might suggest a stolen credit card, even though each individual transaction might appear normal. The collective behavior or pattern highlights the anomaly in this case.

With the emergence of deep anomaly detection methods, recent studies proposed two additional anomaly types to distinguish between the various types of anomalies that deep models aim to detect (Ruff et al., 2021):

- **Sensory (Low-Level) Anomalies:** *Low-level* or *sensory anomalies* refer to the irregularities that occur in the low-level feature hierarchy, such as textures or edges of an image. An example of a low-level anomaly is a fractured texture. Low-level anomaly detection is helpful in detecting defects and artifacts in industrial applications. The recently introduced *MVTecAD* dataset (Bergmann, Fauser, Sattlegger, & Steger, 2019) contains numerous examples of sensory anomalies and defects in industrial applications.
- **Semantic (High-Level) Anomalies:** *High-level* or *semantic anomalies* refer to samples that belong to a different class compared to the normal data. For example, if we train a network to classify cat images as normal samples, any image of an object other than a cat would be considered a semantic anomaly. In this context, the anomaly is determined based on the semantic content or class of the sample rather than low-level features.

It is important to note that both sensory and semantic anomalies might overlap with other types of anomalies. However, it is still essential to distinguish between semantic and sensory anomalies to avoid confusion in our discussions throughout the paper.

3.3. Availability of data labels

To design an appropriate algorithm for anomaly detection, it is crucial to consider the availability of labels. Based on the label availability, AD algorithms can be divided into three settings:

1. **Unsupervised Anomaly Detection:** In this setting, which is arguably the most common in anomaly detection, we assume that only unlabeled data is available for training the model (Hodge & Austin, 2004; Ruff et al., 2021). In the simplified form of unsupervised learning, we commonly assume that the data is noise-free and its distribution is the same as the normal data, e.g. $\mathbb{P} \equiv \mathbb{P}^+$. If noisy data or undetected anomalies are present in the training dataset, these assumptions are violated, hence the developed models are not robust. A more realistic approach can be to assume that the data distribution $\mathbb{P}$ is a mixture of normal data and anomalies with a pollution rate $\eta \in (0, 1)$, e.g. $\mathbb{P} = (1-\eta)\mathbb{P}^+ + \eta\mathbb{P}^-$. In this approach, it is crucial to determine η and make a prior assumption about the distribution of anomalies $\mathbb{P}^-$, which may degrade the method generalization. Overall, the unsupervised settings for anomaly detection gained a great interest in learning commonalities of data from a complex and high-dimensional space without the need to access annotated training samples. Note that the self-supervised learning methods, that are the focus of this paper, can be considered as a subgroup of unsupervised learning techniques.
2. **Semi-supervised Anomaly Detection:** In this setting, we assume that the training dataset is partially labeled and includes both labeled and unlabeled samples. Semi-supervised algorithms are suitable for scenarios where it is costly to annotate the whole data. This setting is also prevalent in anomaly detection because commonly, both labeled and unlabeled data are present, but labeling the data often requires expert knowledge, or in some cases, such as industrial and biomedical applications, anomalies are costly to occur. Incorporating a small set of anomaly samples during training could significantly improve the detection accuracy and maximize the robustness of a model (Kiran, Thomas, & Parakkal, 2018; Min et al., 2018; Ruff et al., 2019), especially compared to the unsupervised learning techniques. However, due to the scarce availability of the labeled abnormal samples, a semi-supervised setting is likely prone to overfitting. Therefore, making the correct assumptions about the distribution of anomalies, i.e. $\mathbb{P}^-$, is crucial for accurately incorporating the labeled anomalies in the training process.
   It is important to note that Some existing papers refer to the task of *Learning from Positive and Unlabeled examples (LPUE)* as semi-supervised learning (Chandola et al., 2009). Note that based on the above definitions, LPUE is an unsupervised learning technique where the entire training data belongs to the normal class. LPUE is commonly used in the literature to benchmark the anomaly detection algorithms using popular datasets, such as CIFAR-10 and MNIST (Golan & El-Yaniv, 2018; Ruff et al., 2018). In this task, the samples of one class of the dataset are deemed normal and are used during the training, and samples of other classes are considered anomalous (Hojjati & Armanfard, 2023). *One-class AD* is another term which is used for referring to the LPUE task.
3. **Supervised Anomaly Detection:** In supervised anomaly detection, we assume that the dataset is fully labeled. When anomalies are easily annotated, it is more beneficial to adopt supervised methods (Feinman, Curtin, Shintre, & Gardner, 2017; Jumutc & Suykens, 2014; Kim, Choi, & Lee, 2015; Lee, Lee, Lee, & Shin, 2018). At this point, it is essential to distinguish between supervised anomaly detection and binary classification problems. One might claim that if the normal and abnormal data are available during the training phase, the problem can be formulated as a supervised binary classification problem and will no longer be an anomaly detection task. However, we should note that, formally speaking, an anomaly is a sample that does not belong to the normal class distribution $\mathbb{P}^+$. The anomaly class includes a broad range of data points that are not accessible/known during the training phase. The common practice anomaly detection is to assume that, in the training phase, there are enough labeled samples from the normal class that can reveal $\mathbb{P}^+$ while the limited available abnormal samples can only partially reveal $\mathbb{P}^-$. Hence, unlike binary classification, which aims to learn a decision boundary separating the two classes, AD seeks to discover the normal class boundaries. Although the supervised settings are more efficient and can achieve higher accuracy, they are rarely used to formulate anomaly detection problems compared to unsupervised and semi-supervised models. This is because, in most real-world applications, it is impossible to describe and have access to all existing anomaly classes.

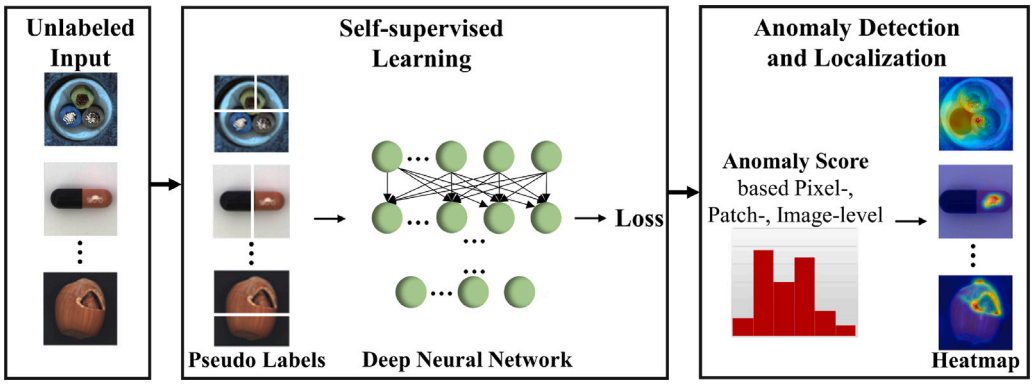


Fig. 2. The overall framework of SSL-AD.

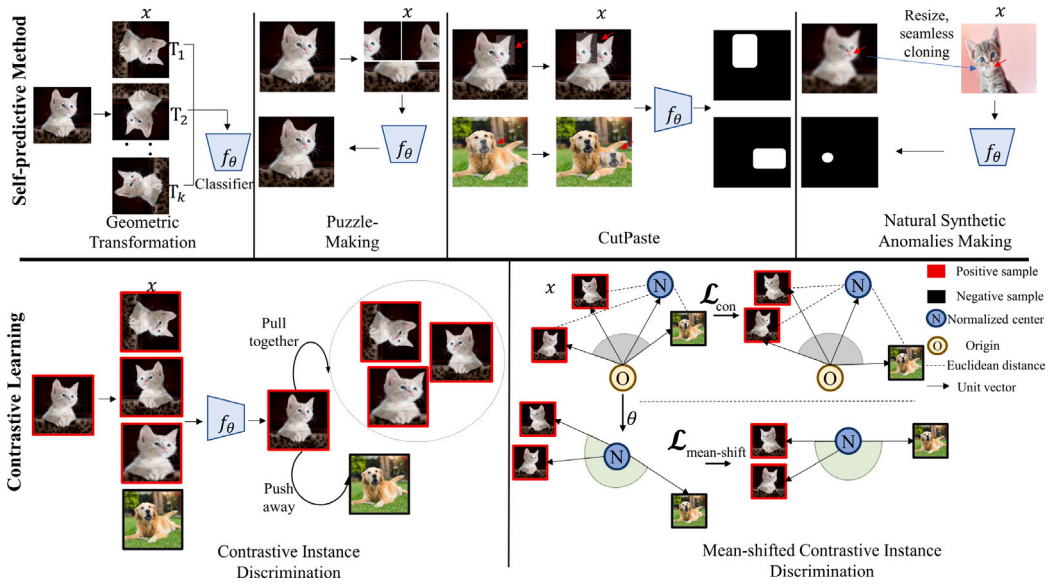


Fig. 3. Several examples of pseudo-label generation processes that are associated with two main categories of SSL-AD. x is the pseudo-labeled input and f_θ is the feature extractor.

4. Self-supervised learning for anomaly detection

Self-supervised learning can leverage large amounts of unlabeled data to learn robust representations of normal behavior, making it a scalable and cost-effective solution for anomaly detection. Nevertheless, SSL algorithms are not inherently suitable for anomaly detection. It is imperative to initially define the anomaly detection problem, followed by applying a relevant SSL algorithm to derive a representation from the input data. This representation should then be mapped into an anomaly score using a suitable function. Subsequently, the algorithm's performance of anomaly detection, and localization tasks (if applicable) should be assessed using appropriate metrics. The general pipeline of SSL-AD is shown in Fig. 2. In this section, we will elaborate on each of these steps in the context of previous SSL-based anomaly detection algorithms.

Tables 1 and 2 provide a summary of the key aspects of SSL anomaly detection papers, including the task they aim to solve, the evaluation metrics used, and how they quantify the anomaly score from the representation. In addition, Fig. 3 illustrates the methodologies employed by each group of methods.

4.1. Problem formulation

Based on the dataset's nature and the availability of data labels, the anomaly detection task can be formulated differently in past studies. The most common formulation is one-class anomaly detection (aka LPUE) (Chen et al., 2020; Golan & El-Yaniv, 2018; Sabokrou et al., 2019), in which one class of the dataset is trained as normal, while the remaining classes are considered abnormal. An example of this task is taking a class of the CIFAR-10 such as *Cat* as normal, and the rest as anomalies. On the other hand, in *multi-class anomaly detection*, multiple classes in the same datasets are considered normal during training, and one or multiple remaining classes are deemed anomalous (Tack et al., 2020; Zhang, Mu, et al., 2022).

4.2. Algorithms

Self-supervised algorithms can learn a proper data representation with the help of a defined pretext supervised task from an unlabeled dataset. The pretext task guides the model to learn a generic representation of the data, which can be helpful for downstream tasks such as classification and anomaly detection. A wide range of proxy tasks and models are proposed in the literature of self-supervised learning. They include but are not limited to, colorization (Larsson et al., 2016), maximization of mutual information between low-level and high-level representations (Hjelm et al., 2019), and predicting geometric transformations (Gidaris et al., 2018). These methods showed promising results in various tasks, such as speech representation learning (Ravanelli et al., 2020), visual feature learning (Jing & Tian, 2021), and healthcare applications (Azizi et al., 2021).

Table 1
Widely-used self-supervised anomaly detection methods.

Category	Method	Task	Anomaly score	Indicator
Self Predictive	GOEM (Golan & El-Yaniv, 2018)	OCAD	Dirichlet Normality	AUC
	NRE (Sabokrou, Khalooei, & Adeli, 2019)	OCAD	Reconstruction Error	EER
	SSL-OE (Hendrycks, Mazeika, Kadavath, & Song, 2019)	OCAD OOD	Rotation Score	AUC
	GOAD (Bergman & Hoshen, 2020)	OCAD	Softmax Probability	AUC
	Puzzle-AE (Salehi, Eftekhar, Sadjadi, Rohban, & Rabiee, 2020)	OCAD	Error Normalization	AUC
	CutPaste citepli2021cutpaste	OCAD	Density Estimator	AUC
	SLA^{2P} (Wang, Qin, et al., 2021)	OCAD	Uncertainty Score	AUC
	NAF-AL (Zhang, Saleeby, et al., 2021)	OCADMCAD	Likelihoods	F1
	DAAD (Zhang, Mu, et al., 2022)	MCAD	Probabilistic scalars with majority voting	F1,AUCACC
	Patch-Based (Tsai, Wu, & Lai, 2022)	OCAD	L_2 Distance	AUC
Contrastive	CLP (Winkens et al., 2020)	OOD	CLP Score	AUC
	CSI (Tack et al., 2020)	OCAD	Cosine similarity, Representation Norm	AUC
	SSD (Sehwag, Chiang, & Mittal, 2021)	OOD	Mahalanobis distance	AUC
	DROC (Sohn, Li, Yoon, Jin, & Pfister, 2020)	OCAD	Normality score	AUC
	MCL (Cho, Seol, & Lee, 2021)	MS AD	Mahalanobis distance	AUC
	NDA (Chen, Xie, et al., 2021)	ND	Reconstruction error	AUC
	Spatial CL (Kim et al., 2022)	OCAD	L_2 Distance	AUC
	Self-Distillation (Rafiee et al., 2022)	OOD	Temperature-Weighted Nonlinear Score	AUC

Self-supervised anomaly detection models vary primarily based on the nature of their proxy tasks. The proxy task is designed to guide the model in learning a representation that is specifically suited for anomaly detection, as opposed to a generic representation learned by an unsupervised model. Research in self-supervised learning has gained unprecedented momentum in recent years and the number of papers in this field has increased exponentially (Gui et al., 2023).

In the past few years, contrastive learning methods have emerged as a significant component of self-supervised learning (Chen et al., 2020). The primary objective of contrastive learning is to develop effective data representations by bringing together different views of the same sample while pushing them apart from other points. To accomplish this, various loss functions have been proposed, such as contrastive loss (Chopra, Hadsell, & LeCun, 2005) and triplet loss (Schroff, Kalenichenko, & Philbin, 2015). Notably, several variants of contrastive learning models have demonstrated impressive accuracy levels comparable to those of fully-supervised models in specific tasks (Chen et al., 2020). Anomaly detection is one of the tasks where SSL algorithms have demonstrated remarkable performance levels that were previously unattainable.

In the latest improvement, contrastive methods such as BYOL (Bootstrap Your Own Latent) (Grill et al., 2020) and SwAV (Swapping Assignments between Views) (Caron et al., 2020) have been developed that further push the envelope by utilizing self-generated negatives through multiple views of the same data. These innovations in contrastive learning have not only improved the efficiency and effectiveness of self-supervised models but have also expanded their applicability across a

Table 2
Summary of widely-used self-supervised anomaly detection methods.

Method	Summary
GOEM	GEOM applies on all the given normal images and encourages learning the features that are useful for detecting novelties.
NRE	Besides learning a reconstruction scheme, AE preserves the local geometric manifold based on NRE that leads to a discriminative neighborhood-guided SSL.
SSL-OE	An auxiliary rotation loss is added to improve the robustness and uncertainty of deep learning models.
GOAD	GOAD uses affine transforms which are suitable for general data. It tries to predict the applied transforms and uses the output of the classifier to detect anomalies.
Puzzle-AE	U-Net solves the puzzled inputs and the robust adversarial training is used as an automatic shortcut removal.
CutPaste	CutPaste augmentation creates local irregular patterns during training and identifies these local irregularity on unseen real defects at the test time.
SLA^{2P}	SLA^{2P} designs a discriminative anomaly score by employing feature-level self-supervised learning and adversarial perturbation.
NAF-AL	It employs data transformations in the SSL setting, and learns the data likelihood by Autoregressive Flow-based Active Learning with Marginal Strategy.
DAAD	It includes a classifier and an adversarial training model. It captures different data distributions and makes an evaluation using the majority voting.
Patch-Based	Incorporates relative feature similarity between patches of varying local distances to enhance information extraction from normal images.
CLP	A simple contrastive training-based approach for OOD detection is proposed. CLP captures the similarity of the inlier and outlier dataset(s).
CSI	A new detection score is introduced in the training phase that contrasts the sample with distributionally-shifted augmentations of itself.
SSD	An outlier detector is based on only unlabeled in-distribution data. SSD uses SSL followed by a Mahalanobis distance in the feature space.
DROC	One-class AD emphasizes the importance of decoupling building classifiers for learning representations.
MCL	MCL can shape dense class-conditional clusters by adding 2 components: class-conditional mask and stochastic positive attraction to boost the performance.
NDA	Negative augmentation generates negative samples closer to normal samples and helps separate normal and abnormal points.
Spatial CL	Incorporates autoencoder in conjunction with contrastive learning to reproduce the original image from cut–paste augmentation.
Self-Distillation	Employs the self-distillation of the in-distribution data, and contrasting against negative examples that are generated through shifting transformations of data.

wide range of domains. Interested readers can refer to Gui et al. (2023) for an up-to-date survey on recent advancements in self-supervised learning research.

Despite their recent success and broad applicability, self-supervised models suffer from several important shortcomings. One of their most significant problems is their computational inefficiency. Compared to a fully-supervised model, they need more time and data to train and get an accuracy comparable to their supervised counterparts.

Inspired by earlier works, we categorize the self-supervised AD models based on their pretext task into two groups (Weng & Kim, 2021):

- **Self-predictive Methods:** These algorithms create the pretext task for each individual sample. Commonly, they apply a transformation to the input and try to either predict the applied transformation or reconstruct the original input. These models are effective even if only *positive* samples, *i.e.* in-distribution (IND) samples, are available. As a result, they do not necessarily require samples from other distributions, also known as *negatives*, during training.
- **Contrastive Methods:** *Contrastive models* define the proxy task on the relationship between pairs of samples. They commonly generate positive views of a sample by applying different geometric transformations. Then, they aim to pull together the positives while pushing them away from the negative ones. In contrastive learning, samples of the current batch other than the anchor sample and its augmentations are considered *negative* while *positive* samples are the ones that are coming from augmentations of the anchor. Technically, contrastive algorithms can also be considered self-predictive. In essence, they also need to learn to predict the transformations to associate the same sample's augmentations with each other. However, the immense advancement of contrastive learning in recent years encouraged us to treat them as a stand-alone category.

Fig. 3 visually illustrates the representation learning process of these two categories. As shown in this figure, unlike self-predictive algorithms, contrastive learning methods incorporate negative samples. This figure also depicts the pseudo-label generation process for different SSL methods. Self-predictive models apply the transformations on positive samples and try to either predict the applied transformation or reconstruct the original input. Contrastive methods, on the other hand, do not explicitly predict the transformations or reconstruct the input and instead aim to distinguish between positive and negative samples. More details on the methods depicted in Fig. 3 are presented in Sections 5 and 6.

In the early stages, the primary focus of algorithms was on image and video anomaly detection. This emphasis was primarily due to the fact that self-supervised representation learning and the related proxy tasks were predominantly developed within the computer vision literature (Chen et al., 2020). Since a significant number of existing works concentrate on image anomaly detection, and this field is well-established, we first discuss the algorithms that were developed for

visual anomaly detection, and subsequently, we will cover the papers that tried to tackle other data types in Section 7.

4.3. Anomaly scoring

Self-supervised models are capable of learning a good feature representation from the input data. However, this representation is not readily useful for anomaly detection. Defining a suitable scoring function to quantify the degree of abnormality from this representation is essential for designing an anomaly detection framework. Previous studies have used a flurry of scoring functions based on the downstream tasks to detect anomalies. For example, two widely used anomaly scores for one-class anomaly detection are normality score and reconstruction error: Normality scores estimate the normality of new samples at the inference time after applying different transformations (Golan & El-Yaniv, 2018; Hendrycks et al., 2019; Li, Sohn, Yoon, & Pfister, 2021; Sohn et al., 2020). Examples of this type of score include the Dirichlet score (Golan & El-Yaniv, 2018) and rotation score (Hendrycks et al., 2019). Reconstruction error, which is typically measured by the Euclidean distance between the original and the reconstructed input, is another category of scoring functions. The assumption behind using this score is that the reconstructed features of anomalies have higher errors than normal samples (Sabokrou et al., 2019; Salehi et al., 2020). For multi-class anomaly detection, scoring functions such as class-wise density estimation (negative Mahalanobis distance) (Sehwag et al., 2021) and data likelihood criterion (Zhang, Saleeby, et al., 2021) were also used. Finally, for tackling the out-of-distribution detection problem, several other measures, including probability-based measures, rotation score (Hendrycks et al., 2019), Confusion Log Probability (CLP) (Winkens et al., 2020), Weighting Softmax Probability (Mohseni, Pitale, Yadawa, & Wang, 2020), and Mahalanobis distance (Sehwag et al., 2021) are used in the self-supervised anomaly detection literature.

4.4. Performance evaluation

To evaluate the performance of an anomaly detector, several criteria are used. In practical applications, the cost of false alarms (type I error) and missed-detected anomalies (type II error) are usually different. Most anomaly detectors define the decision function as

$$\text{Output} = \begin{cases} \text{normal}, & \text{if } Score(x) < \zeta \\ \text{abnormal}, & \text{if } Score(x) \geq \zeta \end{cases},$$

where $Score(x)$ is the anomaly score for new sample x, and the decision threshold ζ is chosen to minimize the costs corresponding to the type I and II errors and to accommodate other constraints imposed by the environment (Field, Tyre, Jonzén, Rhodes, & Possingham, 2004). However, it is common that the costs and constraints are not stable over time or are not fully specified in various scenarios. As an example, consider a financial fraud detector that receives anomaly alarms to investigate potentially fraudulent activities. A detector can only handle a limited number of alarms, and its job is to maximize the number of anomalies containing these alarms based on the precision metric. Meanwhile, an anomaly alarm being wrongly reported can cause a credit card agency placing a hold on the customer's credit card. Thus, the goal is to maximize the number of true alarms, given a constraint on the percentage of false alarms by using the recall metric.

Area Under the Receiver Operating Characteristic (ROC) Curve (AUC or simply AUC) is known for its ability to evaluate the model's performance under a broad range of the decision threshold ζ (Fawcett, 2006). The AUROC curve is an indicator for all sets of precision–recall pairs at all possible thresholds. This makes AUC capable of interpreting the performance of models in various scenarios. As shown in Table 1, most anomaly detection methods use the AUC metric for evaluation. The random baseline achieves an AUC of 0.5, regardless of the imbalance between normal and abnormal subsets, while an excellent model achieves an AUC close to 1, demonstrating the robustness of the model in distinguishing normal from abnormal classes.

5. Self-predictive methods in anomaly detection

Self-predictive methods learn the data embedding by defining the supervised proxy task on a single sample. This approach focuses on the innate relationship between a sample and its own contents or its augmented views. An example of a self-predictive task is masking a portion of an image and trying to reconstruct it using a neural network (Salehi et al., 2020).

The pretext task of self-predictive methods can be categorized into the following groups:

5.1. Transformation-based models

In most self-predictive approaches, the objective is to predict the label of the applied transformation, such as predicting the degree of rotation of an image. In this case, the anomaly score is commonly defined based on the Softmax probabilities of a supervised classifier. Geometric transformations were one of the earliest types of transformations that are used for visual representation learning. Doersch, Gupta, and Efros (2015) showed that predicting the relative position of image patches is a helpful pretext task for improving the representation for object detection. In a later work, Gidaris et al. (2018) used rotation prediction for learning a better representation.

Geometric transformation models first create a self-labeled dataset by applying different geometric transformations to normal samples. The applied transformation is served as the label of each sample. Let $\mathcal{T} = \{T_1, T_2, \dots, T_K\}$ be the set of geometric transformations. The new labeled dataset S can be constructed from the original dataset D as below:

$$S := \{(T_j(x), j) | x \in D, T_j \in \mathcal{T}\}, \tag{1}$$

where the original data point is shown by x. A multi-class network is trained over the dataset S to detect the transformation applied to the sample. During the inference phase, the trained models are applied to the transformed versions of the samples, and the distribution of the Softmax output is used for anomaly detection (Golan & El-Yaniv, 2018). Unlike the Autoencoders and GAN-based methods, the geometric transformation models are discriminative. The intuition behind these models is that the model learns to extract important features of the input by learning to identify the applied geometric transformations. These features can also be helpful for anomaly detection.

The paper by Golan and El-Yaniv (2018) was the first work that used geometric transformation learning for anomaly detection. They named their method as GEOM and showed that it can significantly outperform the state-of-the-art in anomaly detection. They showed that their model can beat the top-performing baseline in CIFAR-10 and CatsVSDogs datasets by 32% and 67%, respectively.

To calculate the anomaly score of a sample from the Softmax probabilities, Golan and El-Yaniv (2018) combined the log-likelihood of the conditional probability of each of the applied transformations:

$$n_S := \sum_{k=1}^{K} \log p(y(T_k(x)) | T_k) \tag{2}$$

Then, they approximated $p(y(T_k(x))|T_k)$ by a Dirichlet distribution:

$$n_S = \sum_{k=1}^{K} (\tilde{\alpha}_k - 1). \log y(T_k(x)). \tag{3}$$

An important issue of GEOM is that the classifier $p(y(T_k(x))|T_k)$ is only valid for samples the network encountered during the training. For other samples which also includes anomalies, $p(y(T_k(x))|T_k)$ can have a very high variance. To address this problem, Hendrycks, Mazeika, and Dietterich (2018) proposed to use some anomalous samples during the training to ensure that $p(y(T_k(x))|T_k) = \frac{1}{M}$ for anomalies. This method, which is also known as Outlier Exposure (OE), formulates the problem as a supervised task which might not be practical for some real-world applications as they do not have access to anomalies.

A significant downside of geometric models is that they only use transformations that are well-suited for image datasets and cannot be generalized to other data types, e.g. tabular data. To overcome this issue, Bergman and Hoshen (2020) proposed a method called GOAD. In GOAD, the data is randomly transformed by several affine transformations $\mathcal{T} = \{T_1, T_2, \dots, T_K\}$. Unlike the geometric transformations, affine transforms are not limited to images and can be applied to any data type. Also, we can show that the geometric transformations are special cases of the affine transform, and the GEOM algorithm is a special case of GOAD. In GOAD, the network learns to map each of the transformations into one hypersphere by minimizing the below triplet loss:

$$L = \sum_i \max\left(\|f(T_m(x_i)) - c_m\|^2 + s - min_{m' \neq m}\|f(T_m(x_i)) - c_{m'}\|^2, 0\right), \tag{4}$$

where $f(.)$ is the network, s is a regularizing term for the distance between hyperspheres, and c_m is the hypersphere center corresponding to the $m-$th transformation.

The above objective encourages the network to learn the hyperspheres with low intra-transformation and high inter-transformation variance. This is to provide a feature space, i.e. the last layer of $f(.)$, in which the different transformations are separated. During the inference phase, the test samples are transformed by all transformations and the likelihood of predicting the correct transform is used as the anomaly score.

Although transformation-based methods showed significant improvement in semantic anomaly detection on datasets such as CIFAR-10, their performance is poor on real-world datasets such as MVTecAD (Salehi et al., 2020). This is because these models can learn high-level features of data by learning the patterns which are present both in the original data and its augmented versions, e.g. rotated instances. However, these algorithms might not be well-suited for sensory-level anomaly detection tasks, e.g. detecting cracks in an object. This is because some types of low-level anomalies, such as texture anomalies, are often invariant to the transformations. To alleviate this issue, several other proxy tasks, that are more suitable for low-level anomaly detection, are proposed.

Another popular transformation-based pretext task is puzzle-solving. It involves creating complex problems and training a model to solve them. By presenting the model with different puzzles, like completing missing parts of an image or predicting what comes next in a sequence, the model learns complex patterns of the data. This improves its ability to understand and apply knowledge across various tasks, including anomaly detection.

Salehi et al. (2020) used the idea of solving the jigsaw puzzle to learn an efficient representation that can be used for pixel-level anomaly detection. Their proposed method, which they named as Puzzle-AE, trains a U-Net autoencoder to reconstruct the puzzled input. The reconstruction objective ensures that the model is sensitive to pixel-level anomalies, while the pretext task of solving the puzzle enables the network to capture high-level semantic information, as shown in Fig. 3. They further boosted the performance of their model by incorporating adversarial training.

5.2. Pseudoanomaly-based methods

Recently, Li et al. (2021) developed a self-supervised method called CutPaste which significantly improves state-of-the-art in defect detection. CutPaste transformation randomly crops a local patch of the image and pastes it back to a different image location. The new augmented dataset is more representative of real anomalies. Thus, the model can be easily trained to identify and localize the local irregularity (shown by the white regions in the black background in Fig. 3). To detect the augmented samples from the un-transformed ones, the objective of the network is defined as follows:

$$\mathcal{L}_{CP} = \mathbb{E}_{x \in \mathcal{X}}\{\mathbb{CE}(g(x), 0) + \mathbb{CE}(g(\mathrm{CP}(x)), 1)\}, \tag{5}$$

where $CP(.)$ is the CutPaste augmentation, $\mathcal{X}$ is the set of normal data, $\mathbb{CE}(.,.)$ is a cross-entropy loss, and g is a binary classifier that can be parameterized by deep networks. In order to calculate the anomaly score from the representation, an algorithm like KDE or GDE can be used.

CutPaste can also learn a patch representation and compute the anomaly score of an image patch by cropping a patch before applying CutPaste augmentation. This facilitates localizing the defective area. In this case, the objective loss function is modified as:

$$\mathbb{E}_{x \in \mathcal{X}}\{\mathbb{CE}(g(c(x)), 0) + \mathbb{CE}(g(\mathrm{CP}(c(x))), 1)\}, \tag{6}$$

where $c(x)$ crops a patch at random location x.

In another related study, Schlüter, Tan, Hou, and Kainz (2021) introduced a novel self-supervised task known as *Natural Synthetic Anomalies (NSA)* for the purpose of detecting and localizing anomalies exclusively using normal training data. Their proposed approach involves generating synthetic anomalies by duplicating patches of various sizes from a source image and incorporating them into a destination image. Specifically, NSA randomly selects a rectangular patch from the source image, resizes it randomly, merges the patch into a different location within the destination image from a distinct source image, and generates a pixel-level mask. The synthetic samples produced by NSA exhibit variations in size, shape, texture, location, color, and other characteristics. In essence, NSA dynamically generates a diverse array of anomalies that offer a more realistic approximation of natural anomalies compared to the samples created by simply pasting patches at different locations. An illustrative example of NSA can be seen in Fig. 3, where a randomly chosen patch from one cat image is seamlessly incorporated into another cat image. Notably, the NSA method surpasses state-of-the-art algorithms in performance across various real-world datasets, including MVTecAD.

5.3. Other methods

Due to the variations of proxy tasks in self-supervised learning, existing SSL-AD methods stretch way beyond the previous three groups. In this section, we show several additional categories of anomaly detection methods based on the anomaly scoring module, as is illustrated in Fig. 4. For instance, the reconstruction-based methodsgenerative methods based on an autoencoder framework, as is indicated in the top row of Fig. 4, are another group of self-supervised algorithms. These methods aim to reconstruct the original input from its masked or transformed version. Denoising autoencoder is a classic example of this approach. In this case, the reconstruction error of the model is often used as the anomaly score. The intuition behind such methods is that every normal sample should be accurately reconstructed, whereas abnormal samples should suffer from the larger reconstruction error. This approach has usually been used in conjunction with other self-supervised methods to boost robustness and performance. For instance, Ho and Armanfard (2023b) incorporated a generative task jointly with graph neural networks and a contrastive learning framework to detect anomalies in brain signals.

The second theme is distribution-based methods (see the second row of Fig. 4). The idea of these methods is to model the distribution of normal data. While normal data is expected to have high likelihood under the probabilistic model, abnormal data should have lower likelihoods. Deep learning methods, such as NAF-AL (Zhang, Saleeby, et al., 2021), were proposed to use extracted deep features jointly with the probabilistic model, hence, shaping the feature space to better satisfy the probabilistic assumptions implicited by the model.

The third paradigm is classification-based methods, as is shown in the last row of Fig. 4. This approach aims to learn a good feature space

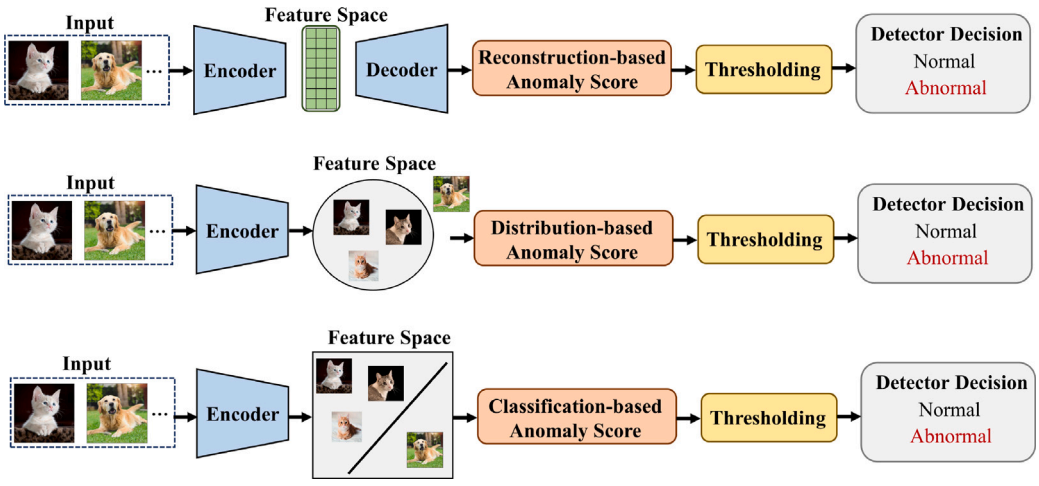


Fig. 4. Additional categories of Self-predictive methods based on Anomaly Scoring are Reconstruction-based, Distribution-based, and Classification-based methods.

to separate the normal and abnormal regions. GOAD is an example of this category and has shown the state-of-the-art performance on anomaly detection using a classification-based approach, i.e., GOAD trains a classifier on a set of random proxy tasks.

Not limited to the three above categories, another important group of SSL-AD methods are hybrid models which combine SSL with supervised learning, recently showing significant improvements over the state-of-the-art. Even though self-predictive models showed promising results, their performance is still significantly poorer than fully-supervised models in out-of-distribution detection. However, some recent studies (Hendrycks et al., 2019) hinted that using SSL models in conjunction with supervised methods can improve the robustness of the model in different ways. Therefore, even in cases where we have access to anomaly data and labels, using self-supervised proxy tasks can enhance the performance of the anomaly detector.

6. Contrastive methods

6.1. Contrastive learning: Basics

The primary objective of contrastive self-supervised learning is to learn a feature space or a representation in which the positive samples are closer together and are further away from the negative points (Sadeghi, Hojjati, & Armanfard, 2023). Empirical evidence shows that contrastive learning models such as SimCLR (Chen et al., 2020) and MoCo (He, Fan, Wu, Xie, & Girshick, 2020) are particularly efficient in computer vision tasks. SimCLR, one of most popular recent contrastive learning algorithms, learns representations by maximizing the agreement between different augmented versions of the same image while repelling them from other samples in the batch. Each image x_i from randomly sampled batch $\mathcal{B} = (\{x_i, y_i\})_{i=1}^{N}$ is augmented twice, producing an independent pair of views $\{\hat{x}_{2i-1}, \hat{x}_{2i}\}$, and augmented batch $\hat{\mathcal{B}} = \{(\hat{x}_i, \hat{y}_i)\}_{i=1}^{2N}$, where the labels of augmented data $\{\hat{y}_{2i-1}, \hat{y}_{2i}\}$ are equal to the original label y_i. By performing independent transformation T and T′ drawn from a pre-defined augmentation function pool $\mathcal{T}$, the augmented pair of views $\{\hat{x}_{2i-1} = \mathrm{T}(x_i), \hat{x}_{2i} = \mathrm{T}'(x_i)\}$ are then generated. Next, $\{\hat{x}_{2i-1}, \hat{x}_{2i}\}$ are passed sequentially through an encoder and a projection head to yield latent vectors $\{z_{2i-1}, z_{2i}\}$. SimCLR learns the representation by minimizing the following loss for a positive pair of examples (m, n):

$$l(m,n) = -\log \frac{\exp(sim(z_m, z_n)/\tau)}{\sum_{i=1}^{2N} \mathbf{1}_{\{i \neq m\}} \exp(sim(z_m, z_i)/\tau)} \tag{7}$$

where $sim(z_m, z_n)$ represents the cosine similarity between the pair of latent vectors (z_m, z_n), $\mathbf{1}_{i \neq m}$ is an indicator function which is equal to 1 if $i \neq m$ and zero otherwise, and τ indicates the temperature hyperparameter which determines the degree of repulsion. The final objective is to minimize the contrastive loss, defined in (8), over all positive pairs in a mini-batch:

$$\mathcal{L}_{SimCLR} = \frac{1}{2N} \sum_{i=1}^{N} \Big[l(2i-1, 2i) + l(2i, 2i-1) \Big]. \tag{8}$$

6.2. Contrastive learning for anomaly detection

Contrastive learning models established themselves as powerful representation learning tools. Still, they face crucial challenges for anomaly detection. Most widely-used contrastive learning algorithms, such as SimCLR and MoCo, need negative samples to operate. However, we either only have access to the samples from one class in many anomaly detection tasks, or the distribution of classes is highly imbalanced. In addition, the learned representation is not readily suitable for the anomaly detection task, and we need to define a proper anomaly score.

Despite these challenges, several contrastive anomaly detection models have emerged in the recent years. We illustrate a contrastive learning paradigm based distribution for anomaly detection in Fig. 5. The CSI method proposed by Tack et al. (2020) was the first attempt for using contrastive learning in anomaly detection. The CSI method is based on the idea of instance discrimination which considers every data point as a separate class and negative relative to other samples in the dataset (Wu, Xiong, Yu, & Lin, 2018). This idea is proven to be practical in visual representation learning for classification, but its performance in anomaly detection is unexplored (Chen et al., 2020). They also showed that if specific transformations are used for generating negative samples from a given point, the learned representation can be more appropriate for anomaly detection. These distribution-shifting transformations can be denoted by a set as $\mathcal{S}$. In contrast to SimCLR, which considers augmented samples as positive to each other, CSI attempts to consider them as negative if the augmentation is drawn from $\mathcal{S}$. A significant conclusion of the CSI method is that although using the shifted transformations does not improve and even in some cases hurts the performance of the representation in other downstream tasks such as classification, it can improve the performance for anomaly detection.

If we denote the set of shifting transformations by $\mathcal{S} = \{S_0 = I, S_1, \ldots, S_{K-1}\}$ with I being the identity function and K different (either random or deterministic) transformations, the CSI loss can be written as:

$$\mathcal{L}_{con-SI} := \mathcal{L}_{SimCLR}\Big(\bigcup_{S \in \mathcal{S}} \mathcal{B}_S; \mathcal{T} \Big) \tag{9}$$

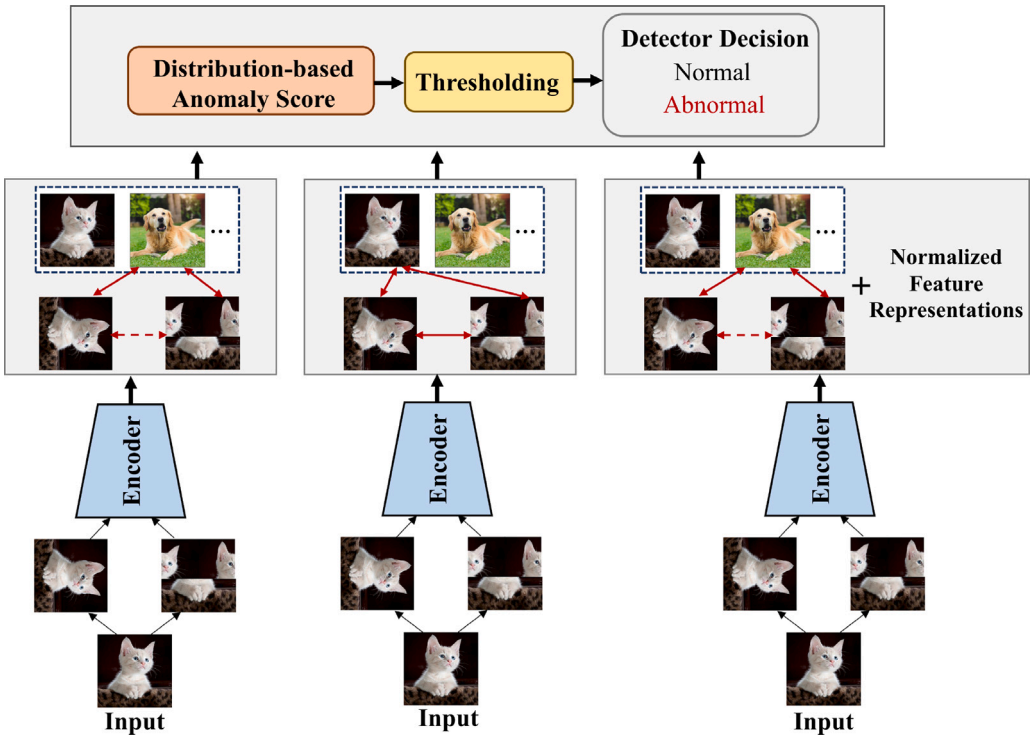


Fig. 5. A contrastive learning paradigm for anomaly detection based on the process of contrasting samples. From left to right, there are the processes of contrasting augmented samples from different classes, contrasting augmented samples from the same class and contrasting augmented samples from different classes with normalized feature representations, respectively. Anomaly Scoring module based distribution is used. The red dashed line represents attraction, whereas repulsion is represented by the red solid line.

in which $B_S := \{S(x_i)\}_{i=1}^{B}$. In simpler terms, the $\mathcal{L}_{con-SI}$ is essentially the same as the SimCLR loss, but in the con-SI, the augmented samples are considered negative to each other.

In addition to discriminating each shifted instances, an auxiliary task is added with a Softmax classifier $p_{cls-SI}(y^S|x)$ that predicts which shifting transformation $y^S \in S$ is applied for a given input x_i. The classifying shifted instances (cls-SI) loss is defined as below:

$$\mathcal{L}_{cls-SI} := \frac{1}{2B}\frac{1}{K}\sum_{S\in S}\sum_{\hat{x}_S\in\hat{B}_S} -\log p_{cls-SI}(y^S = S|\hat{x}_S) \tag{10}$$

The final loss of CSI is then defined as:

$$\mathcal{L}_{CSI} := \mathcal{L}_{con-SI} + \lambda\mathcal{L}_{cls-SI} \tag{11}$$

The authors of the CSI empirically showed that the norm of the representation $\|z(x)\|$ is indeed a good anomaly score, where z is the representation vector and $\|\cdot\|$ denotes the second norm. This can be explained intuitively by considering that the contrastive loss increases the norm of the in-distribution samples to maximize the cosine similarity of samples generating from the same anchor. Consequently, during the test time, in-distribution samples are mapped further from the origin of the z space, while the representation of other data points, i.e. anomalies, have a smaller norm hence are closer to the origin. This is an important observation as it helps to solve the problem of defining the anomaly score on a representation that is learned in an unsupervised fashion. The authors also found that the cosine similarity to the nearest training point in $\{x_m\}$ can be another good anomaly score. They defined the score of their model as a combination of these two metrics as below:

$$s_{con}(x;\{x_m\}) := \max_m \text{sim}(z(x_m), z_x).\|z(x)\| \tag{12}$$

where z_x is the representation vector of the test sample x and $z(x_m)$ is the closest representation vector in the training set.

6.3. Improving CL: Masked contrastive learning

The CSI algorithm shows that the task-agnostic representation learned through contrastive learning is suitable for anomaly detection. However, a task-specific approach can be more suitable for anomaly detection. (The task may be defined as the AD task itself or another downstream task such as data classification.) The contrastive models, such as SimCLR, are quite helpful in learning a representation for individual data points. They can also learn separable clusters for each class without having access to any labels. However, the resulting clusters may have blurry boundaries, and they commonly require fine-tuning for the downstream tasks.

To overcome this obstacle, Cho, Seol, and Lee (2021) developed a contrastive model which is tailored for anomaly detection. Their model, which is called Masked Contrastive Learning (MCL), modifies the degree of repulsion based on the labels of the data points. In vanilla SimCLR, all other batch samples, regardless of their class label, are considered negative relative to the anchor sample and are repelled with equal magnitude. However, in MCL, the repelling ratio is defined by the following class-conditional mask (CCM):

$$\text{CCM}(m,n) = \begin{cases} \alpha & \text{if } \bar{y}_m = \bar{y}_n \\ \frac{1}{\tau} & \text{if } \bar{y}_m \neq \bar{y}_n, \end{cases} \tag{13}$$

where $0 < \alpha < \frac{1}{\tau}$. Basically, CCM adjusts the temperature τ for the same labeled views to a smaller value of α. This means that if the negative sample has the same class as the anchor, it is repelled with less magnitude compared to other data points. The SimCLR loss function is modified according to this mask as follows:

$$\mathcal{L}_{CCM} = \frac{1}{2N}\sum_{i=1}^{N}\Big[l_{CCM}(2i-1,2i) + l_{CCM}(2i,2i-1)\Big], \tag{14}$$

$$p_{CCM}(m,n) = \frac{\exp(sim(z_m,z_n)/\tau)}{\sum_{i=1}^{2N}\mathbf{1}_{\{i\neq m\}}\exp(sim(z_m,z_i)).\text{CCM}(m,i)}, \tag{15}$$

$$l_{CCM}(m,n) = -\log p_{CCM}(m,n), \tag{16}$$

Although the proposed mask leads to a finer-grained representation space, the repulsive nature of the loss function may lead to the formation of scattered clusters. To prevent this phenomenon, the MCL algorithm stochastically attracts each sample to the instances with the same class label.

To further improve the MCL model in Cho, Seol, and Lee (2021), an auxiliary classifier that predicts the applied transformation is also employed. The masking function is then modified based on the label of sample and its transformations. The repelling ratio is then smaller for the samples that simultaneously have the same class label and transformation labels, compared to the samples with the same class but different transformation labels. A sample with the latter property repels with a smaller magnitude than the negative points.

To score the anomalies in Cho, Seol, and Lee (2021), the Mahalanobis distance (Mahalanobis, 1936), shown in (17), is employed.

$$MD(x) = (z_x - \mu)^T \Sigma^{-1} (z_x - \mu), \tag{17}$$

where z_x is the representation of x, μ is the sample mean, and Σ is the sample covariance of features of the in-distribution training data. The Mahalanobis distance is a standard metric for scoring anomalies from their representation. It does not require any labeled data that makes it a common choice for many anomaly detection algorithms. In addition to this distance, the score of the auxiliary classifier is used to boost the model's robustness.

6.4. One-class contrastive anomaly detection

Contrastive models are also used in conjunction with one-class models for anomaly detection. One-class classifiers are one of the most widely used models in anomaly detection. They can detect anomalies after learning from a single class of examples. Sohn et al. (2020) employed a two-stage framework for detecting anomalies using self-supervised learning models. In this framework, an SSL-based neural network is used to learn the representation of the input. A one-class classifier, such as OCSM or KDE, is applied to the learned representation to detect anomalies. The two-stage framework eliminates the need for defining an anomaly score and, as is empirically demonstrated in the paper, it can outperform other state-of-the-art methods.

Despite their promising empirical results, one-class classifiers suffer from a critical problem known as catastrophic collapse. This phenomenon happens when the network converges to the trivial solution of mapping all the inputs to a single point regardless of the input sample value x, i.e. $\phi(x) = c$ where $\phi(\cdot)$ denotes the network output. This trivial solution is obtained when minimizing the center-loss defined as $\mathcal{L} = \|\phi(x) - c\|^2$ (Reiss, Cohen, Bergman, & Hoshen, 2021; Ruff et al., 2018). The features that the network learns in such case are uninformative and cannot be used for distinguishing anomalies from normal data. This issue is also known as "hypersphere collapse".

To overcome the hypersphere collapse problem, Reiss and Hoshen (2021) proposed a new loss function, called Mean-shifted contrastive loss (MSCL). Unlike the conventional contrastive loss, where the angular distance is computed relative to the origin, MSCL measures the angular distance relative to the normalized center of the extracted features. An example of MSCL is shown in Fig. 3. Formally, for a sample x, the mean-shifted representation is defined as:

$$\theta(x) = \frac{\phi(x) - c}{\|\phi(x) - c\|},$$

The mean-shifted contrastive loss is then given by:

$$\begin{aligned}\mathcal{L}_{MSCL}(x', x'') &= \mathcal{L}_{CONS}(\theta(x'), \theta(x'')) \\ &= -\log \frac{exp((\theta(x').\theta(x''))/\tau)}{\sum_{i=1}^{2N} \mathbf{1}[x_i \neq x'].exp((\theta(x').\theta(x_i))/\tau)},\end{aligned} \tag{18}$$

where $\mathcal{L}_{CONS}$ is the typical contrastive loss for a positive pair, shown in SimCLR (Chen et al., 2020), and x, x'' are the two augmentations of the input x.

One limitation of the MSCL loss is that it implicitly encourages the network to increase the distance of features from the center. Because of this, normal data lie in a region far away from the center. To solve this issue, the loss function is modified by adding the angular center loss, which shrinks the distance of normal samples from the center. Reiss et al. (2021) showed that the overall loss, which is a combination of the MSCL and the angular losses, can achieve a better training stability and higher accuracy in anomaly detection than the regular center-loss.

6.5. Contrastive learning for out-of-distribution detection

Parallel to Tack et al. (2020), Winkens et al. (2020) developed a contrastive model for detecting out-of-distribution instances. They evaluated their approach on several benchmark OOD tasks and showed that contrastive models are also capable of OOD. The paper's key idea is that a fully supervised model might not be able to capture the patterns that can be useful for out-of-distribution detection. However, using contrastive learning techniques, the model learns high-level and task-agnostic features that can also help detect OODs. When we combine these techniques with the supervised learning techniques, the resulting model can learn more reliable features for both semantic classification and OOD detection.

In another similar work, Sehwag et al. (2021) explored the applicability of contrastive self-supervised learning for out-of-distribution (OOD) and anomaly detection from unlabeled data, and proposed a method called SSD. They also extended their algorithm to work with labeled data in two scenarios: First is the scenario in which it is assumed that there are a few labeled out-of-distribution samples (i.e. a k-shot learning setting where k is set to 1 or 5), and the second scenario is the case in which labels of the in-distribution data are provided during the training phase.

In SSD (Sehwag et al., 2021), the SimCLR is used to learn the representation and the Mahalanobis Distance is incorporated to detect anomalies. For the cases where the labeled data is present, the authors suggested using the SupCon loss, defined in (19), which is a supervised variant of the contrastive loss (Khosla et al., 2020), to have a more effective selection of the positive and negative samples for each image. In SupCon, samples from the same class are treated as positive and other samples as negatives.

$$\mathcal{L}_{SupCon} = \frac{1}{2N} \sum_{i=1}^{2N} -\log \frac{\frac{1}{2N_{y_m}-1} \sum_{i=1}^{2N} \mathbf{1}(i \neq m)\mathbf{1}(y_i = y_m) e^{u_m^T u_i/\tau}}{\sum_{i=1}^{2N} \mathbf{1}(i \neq m) e^{u_m^T u_i/\tau}}, \tag{19}$$

where N_{y_m} refers to the number of images with label y_m in the batch, and $u_i = \frac{h(f(x))}{\|h(f(x))\|_2}$ with a projection head $h(\cdot)$ and an encoder $f(\cdot)$. Using SupCon loss yielded better performance compared to the contrastive loss throughout their experiments for the OOD detection from a labeled dataset. Overall, Sehwag et al. (2021) showed that the contrastive approach can outperform other methods in OOD detection in both labeled and unlabeled settings.

In summary, recent papers suggest that the representation that is learned through self-supervised learning is indeed very useful for anomaly detection. An interesting observation is that even a simple scoring function such as the norm of the representation $\|z\|$ can be used for detecting anomalies from the representations. This can be justified because, in CL-based models, the normal data is spread out on a hypersphere. This property can help to define the anomaly score as the distance of the representation from the center. A smaller distance means a higher probability of the point belonging to the anomaly class.

7. Self-supervised anomaly detection beyond images

In recent years, there has been a growing interest in extending self-supervised anomaly detection techniques beyond image data. While the majority of early research in anomaly detection focused on image and

Table 3
Self-supervised anomaly detection for non-image data.

Data type	Paper	Type	Idea
Audio	Giri et al. (2020)	Self-Predictive	Machine ID Classification
	Kim, Ho, and Kang (2021)	Self-Predictive	Machine ID Classification
	Hojjati and Armanfard (2022)	Contrastive	Pitch Shift, Fade In/Out, Time-Stretch, etc.
	Guan, Xiao, Liu, Zhu, and Wang (2023)	Contrastive	Machine ID Classification Contrastive Pretraining
	Zeng et al. (2023)	Contrastive	Joint Generative/Contrastive Representation Learning
	Bai, Chen, Wang, Ayub, and Yan (2023)	Self-Predictive	Time Masking and Machine ID Classification
Time-Series	Carmona, Aubet, Flunkert, and Gasthaus (2022)	Self-Predictive	Anomaly Injection
	Ho and Armanfard (2023b)	Contrastive	Graph Contrastive Learning Masked Sensor Reconstruction
	Hojjati, Sadeghi, and Armanfard (2023)	Contrastive	Contrastive Learning Between Time Blocks
	Wang et al. (2023)	Contrastive	Joint contrastive and one-class classification
	Jeong, Yang, Ryu, Park, and Kang (2023)	Self-Predictive	Synthetic Anomaly Injection
	Zhang, Zhao, et al. (2022)	Self-Predictive	Intra-Sample Prediction Task
	Fu and Xue (2022)	Self-Predictive	Masked Data Reconstruction
	Jiao, Yang, Song, and Tao (2022)	Contrastive	Pseudo-Negative Generation
	Huang, Shen, et al. (2022)	Self-Predictive	Detection the Downsampling Resolution
Graph	Liu, Li, et al. (2021)	Contrastive	Sub-graph Contrastive Learning
	Zheng et al. (2021)	Contrastive	Sub-graph Contrastive Learning and Node Reconstruction
	Duan et al. (2022)	Contrastive	Graph Views with Node- and Sub-graph-level Contrastive Learning
	Chen et al. (2022)	Contrastive	Node-level Supervised Contrastive Learning
	Xu, Huang, Zhao, Dong, and Li (2022)	Contrastive	Graph-level Supervised Contrastive Learning and Reconstruction
	Zheng et al. (2022)	Contrastive	Graph-level Few-shot Contrastive Learning
	Huang, Pei, Menkovski, and Pechenizkiy (2022)	Self-Predictive	Node- and Graph-level based Hop Count Prediction
	Liu, Pan, et al. (2021)	Contrastive	Edge-level Contrastive Learning in Dynamic Graphs
	Luo et al. (2022)	Contrastive	Node- and Graph-level Contrastive Learning
	Ho and Armanfard (2023b)	Contrastive	Node- and Sub-graph-level Contrastive Learning and Reconstruction
Other	Qiu, Pfrommer, Kloft, Mandt, and Rudolph (2021)	Self-Predictive	Trainable Transformations
	Manolache, Brad, and Burceanu (2021a)	Self-Predictive	Text Anomaly Detection
	Shenkar and Wolf (2022)	Contrastive	Tabular Data Anomaly Detection

video data, the need to detect anomalies in various other data types, such as text, audio, and time series, has become increasingly apparent. In this section, we delve into the advancements made in self-supervised anomaly detection methods that specifically target non-image data.

A crucial aspect of self-supervised learning methods is the selection of data-specific augmentations and proxy tasks. In the context of non-image self-supervised anomaly detection, a primary focus lies in defining a set of augmentations and proxy tasks that are effective for detecting anomalies. Inspired by image anomaly detection models, many algorithms have sought to adapt and extend these techniques for different data types. Table 3 summarizes the important papers in this field. In the following subsections, we explore various data types and their corresponding algorithms, shedding light on their augmentations and proxy tasks.

7.1. Audio anomaly detection

Audio data plays a significant role in various applications, including speech recognition, environmental monitoring, and acoustic anomaly detection. The detection of audio anomalies has been a longstanding research challenge. However, more recently, self-supervised methods have emerged as successful approaches for addressing this task. In the realm of audio data, much like in images and videos, the outcomes of augmenting transformations can be evaluated qualitatively. As a consequence, the literature has already established a robust set of

positive and negative transformations that have proven effective. These include well-known techniques such as noise injection, pitch shifting, and fade in/fade out, among others. These established transformations have been used in conjunction with the ideas from self-supervised visual anomaly detection to develop new models for acoustic data. Another helpful aspect of audio data is that their spectrogram, which is an essential tool in anomaly detection, can be used as input to computer vision models such as CNNs. As a result, they are compatible with existing image self-supervised representation learning tools.

Giri et al. (2020) was one of the first studies that adapted the idea of self-supervised learning for detecting abnormal machine conditions. They have incorporated augmentations such as linearly combining the audio and warping the spectrograms in order to learn a representation which is suitable for anomaly detection. Their research demonstrated that their proposed method surpasses existing baselines by a significant margin. In another similar work, Kim et al. (2021) introduced an innovative framework for acoustic anomaly detection that incorporates the concept of self-supervision. In this algorithm, accurately identifying the machine ID associated with a given sound is defined as the proxy task. Additionally, they leveraged phase continuity information and employed the complex spectrum as input to their model. During the inference phase, any data that the model was unable to classify correctly with the corresponding machine ID has been deemed an anomaly. The experimental evaluations conducted in the paper demonstrated that the utilization of a simple proxy task yielded impressive results, significantly enhancing the model's ability to detect anomalies.

For the first time, Hojjati and Armanfard (2022) introduced a contrastive framework for acoustic anomaly detection. They defined a comprehensive set of transformations, such as time and frequency masking, pitch shift, and noise injection, specifically designed for audio data. These transformations were utilized to create positive and negative pairs for training a contrastive learning algorithm. They have shown that this approach significantly outperforms other existing methods and highlighted the remarkable improvement that could be achieved through contrastive learning in acoustic data. Following this work, Guan et al. (2023) proposed a method that combines contrastive learning with the proxy task of machine ID detection to improve accuracy.

These advancements have shown promise in detecting anomalous sounds, such as abnormal environmental sounds or audio events in surveillance systems.

7.2. Time-series anomaly detection

Time series data arises in a wide range of domains, including finance, manufacturing, and healthcare. Detecting anomalies in time series is crucial for identifying unusual patterns or behaviors. Self-supervised learning techniques have been leveraged to capture temporal dependencies and detect anomalies in time series data.

Unlike images, videos, and audio data, defining suitable augmentations for time-series data is an exceptionally challenging task that heavily relies on the target application and characteristics of the data. Despite this inherent difficulty, researchers have proposed several ideas in recent years to adapt the self-supervised learning framework to time series. One particularly popular approach, which can be applied to a wide range of time series, involves injecting synthetic anomalies and training the network to distinguish them from positive samples. In an early attempt, Carmona et al. (2022) developed *Neural Contextual Anomaly Detection* (NCAD), which could learn the boundary between normal and abnormal samples by injecting pseudo-negative samples during training. To generate these anomalies, they drew inspiration from Hendrycks et al. (2019), and replaced segments of the original time series with values obtained from another time series. To further enhance the diversity of the negative set, they also included synthetic point anomalies. A similar concept was employed by Jiao et al. (2022) to generate synthetic anomalies and train a representation using contrastive learning, which enables the discrimination between positive and negative samples. Very recently, Jeong et al. (2023) used the idea of synthetic anomaly injection in conjunction with the self-attention mechanism to detect abnormal sequences with high accuracy.

Another widely applicable and popular idea is the masking of a segment of the time-series data and training the network to reconstruct it. This concept has been successfully employed in image and audio anomaly detection. Notably, Fu and Xue (2022) demonstrated that this approach could also be effectively utilized for learning efficient representations in time-series data. The underlying assumption behind this idea is that by learning to reconstruct the masked segment, the network will learn the patterns that are present in normal data. In the case of multivariate time series, a possible implementation involves masking the data of one time series and using the data from other entities to reconstruct or predict it (Ho & Armanfard, 2023b; Zhang, Zhao, et al., 2022). This allows the model to capture the dependencies and relationships between different entities within the time-series data. Additionally, Ho and Armanfard (2023a) developed a self-supervised method coupled with different masking strategies to detect anomalies when the training data are contaminated with noise.

A notable trend in time-series anomaly detection involves leveraging temporal information of the data. This approach aims to capture meaningful patterns and enhance the learning of efficient representations. For example, Huang, Shen, et al. (2022) demonstrated that predicting the downsampling resolution of the data can significantly contribute to learning effective representations from time series. By incorporating the downsampling resolution prediction task, the network is encouraged to understand the underlying temporal structure and capture essential features at different resolutions. This enables the model to develop a comprehensive understanding of the time-series data, leading to improved anomaly detection performance. Additionally, researchers such as Hojjati et al. (2023) have utilized temporal adjacency information to generate positive and negative pairs for training contrastive learning models. This approach enhances the model's ability to capture contextual information and detect anomalies by comparing similar and dissimilar pairs of temporal instances.

In conclusion, self-supervised learning techniques offer promising avenues for time-series anomaly detection. The injection of synthetic anomalies, along with methods such as contrastive learning and resolution prediction, enables the network to learn efficient representations and distinguish between normal and abnormal sequences.

7.3. Graph anomaly detection

Following the great success of SSL in the image/signal/text domains, very recently, SSL has gained significant attention in graph-structured data. A graph is a representation of a network, consisting of nodes that represent entities (e.g., objects, users, sensors) and edges that represent the interactions between entities. These interactions/relationships between nodes are known as structural dependencies and are expressed by the adjacency matrix (aka a square matrix) (Liu et al., 2022). Each row and column of the matrix is associated with a node in the graph. The non-zero value in the entry of the matrix indicates whether there is an edge between two nodes. Given this unique property, graphs are different from other domains since the samples (nodes) are dependent on each other in the graph, while the samples in images or texts are independent. Due to such dependencies, it is therefore non-trivial to adopt pretext tasks designed for images or texts directly to graphs.

Many recent SSL methods have provided well-designed pretext tasks based contrastive learning that are applicable for graphs to deal with graph anomaly detection, the task of detecting anomalies (e.g., anomalous nodes, edges, sub-graphs) in *static* graphs. Note that in a static graph, oftentimes seen in social networks, the sets of nodes/edges and their features, as well as the adjacency matrix are fixed. Liu, Li,

et al. (2021) proposed a local sub-graph-based sampling, which pays attention to the relationship between a node and its neighbors in a static graph, to select contrastive pairs. A pair consists of a node and its neighboring sub-graph. A positive pair composes of a target node and its neighboring sub-graph, while a negative pair consists of a node and its corresponding sub-graph. Note that a target node can be any node in a graph, a selected node in a negative pair is different from the target node selected in a positive pair, hence there is mismatching between the target node and the sub-graph in a negative pair. A contrastive-based module is designed to estimate the matching between the target node and sub-graphs in contrastive pairs and would assign the abnormality level for every node.

Zheng et al. (2021) also aimed to compute the level of abnormality of every node in a static graph by designing an effective graph view sampling technique. Given a target node, two positive sub-graphs are sampled, and two negative sub-graphs are sampled randomly and guaranteed that they are different from positive sub-graphs. They designed two pretext tasks, one is to determine the mismatching between the target node and its sub-graphs in contrastive pairs as similar to Liu, Li, et al. (2021), the other is to reconstruct the target node's features based on surrounding nodes in positive sub-graphs. As a result, by taking advantage of multiple pretext tasks, Zheng et al. (2021) showed better detection performance on anomalous nodes than Liu, Li, et al. (2021).

Not limited to sub-graph-level sampling, Duan et al. (2022) showed the effectiveness of combining various contrastive pair sampling strategies. Given the original graph input as the first view, they adopted edge modification to generate the second view of the graph. For each view, they combined node–subgraph, node–node and subgraph–subgraph sampling techniques. The first two techniques can capture sub-graph- and node-level anomalous information in each view, while the latter focuses on more global anomalous information between two views. They showed that a diversity of sampling techniques helps to learn more representative and intrinsic graph embeddings, which could further improve the anomaly detection performance.

While the above studies are unsupervised graph anomaly detection methods, i.e., no annotated labels are available in the training phase, several studies leveraged prior human knowledge on graph anomalies. For example, Chen et al. (2022) took advantage of prior human knowledge, hence, they designed a contrastive loss and trained the model in a supervised manner, i.e., labeled normal and abnormal nodes are respectively treated as positive and negative samples. Xu et al. (2022) also used human knowledge for helping the detection performance, but the way of building their contrastive pairs is different from Chen et al. (2022). Given the actual anomalous static graph, they augmented a new graph by a knowledge modeling technique, then fed both original and augmented graphs to a Siamese graph neural network such that both graphs are encoded into the same latent space, making it feasible to contrast original and augmented graphs. After encoding, they designed a contrastive loss that is integrated with the human knowledge of anomalies, i.e., the contrastive loss would guide the encoder to differently represent the nodes in the original graph and the nodes in the augmented graph. Zheng et al. (2022) also verified the effectiveness of having prior human knowledge in graph anomaly detection by proposing to use few anomalous samples in the training phase. This technique is known as few-shot supervised learning that could enrich the supervision signals for the model, hence, the detection accuracy could be improved.

As is seen from the aforementioned studies, most of techniques used the local context of graphs (i.e., the sub-graph knowledge) and adopted contrastive learning, however, Huang, Pei, et al. (2022) showed that using only local information is insufficient to effectively detect anomalies. More specifically, they designed a self-predictive framework for hop count (aka the shortest path length between pairs of nodes) prediction task, which considers both local and global information. The intuition behind hop counts based on local and global information is that since node-level anomalies are different normal nodes at both the feature- and adjacency matrix-levels, the distance between an anomalous node and its surrounding nodes should be larger than that between a normal node and its neighboring nodes. Hence, computing hop counts based on both local and global information can be useful to construct an anomaly indicator.

SSL with well-designed pretext tasks has shown a capability to handle complex structural dependencies and detect graph anomalies in static graphs. However, detecting anomalous graph objects raises an even more difficult problem in a *dynamic* graph (aka a graph set), which consists of consecutive temporal graphs indexed in time, hence, the feature sets of nodes/edges and adjacency matrices change overtime. Time-series signals, edge streams in social networks, and videos are some of the examples that can be converted to dynamic graphs (Ho, Karami, & Armanfard, 2023). Several studies have shown the potential of SSL to detect anomalies in dynamic graphs. For example, Liu, Pan, et al. (2021) aimed to detect edge-level anomalies at different time steps in an edge stream by designing a dynamic graph transformer-based contrastive learning. Positive edges are sampled from the normal training set while negative edges are randomly sampled based on a random sampling technique and are guaranteed that these negative samples are different from positive samples.

Other additional examples have demonstrated the ability of SSL in dynamic graphs constructed from different data modalities. For example, Luo et al. (2022) aimed to detect anomalies in molecular networks, protein networks and social networks. They first constructed dynamic graphs for these networks. Then, they leveraged contrastive learning to capture both node-level and graph-level representations by a dual-graph encoder, and aimed to detect graph-level anomalies. Ho and Armanfard (2023b) aimed to effectively construct a graph set for time-series signal data, and then detect node-level and sub-graph-level anomalies in constructed graphs. To do so, they utilized the reconstruction-based and contrastive-based SSL pretext tasks to effectively capture the local sub-graph information in graphs.

In conclusion, SSL have yielded promising results for detecting anomalous graph objects at the node-, edge-, sub-graph- and graph-levels in both static and dynamic graphs. Using the knowledge of the features sets of nodes/edges, the adjacency matrices, the local and global information in graphs, and more importantly designing a diversity of effective graph augmentation techniques for pretext tasks would significantly improve the method's detection performance.

7.4. Anomaly detection in other non-image data types

Beyond Graphs, audio, and time series data, self-supervised anomaly detection techniques have also been successfully applied to other data types. In particular, Shenkar and Wolf (2022) introduced an innovative contrastive learning algorithm specifically designed for tabular data. Their approach involved incorporating the concept of feature masking as a proxy task. During the training process, the model learns to create a mapping that maximizes the mutual information (MI) between the original samples and the masked features. To identify anomalies, the contrastive loss itself is directly used as the anomaly score. The findings of this study demonstrated the efficacy of self-supervised learning in tabular anomaly detection.

Another area that has recently garnered attention is text anomaly detection using self-supervision. Manolache, Brad, and Burceanu (2021b) introduced a novel proxy task called *Replaced Mask Detection* (RMD), which involves two steps: (I) Masking a particular word in the input, and (II) Replacing the masked word with an alternative. The model is trained to differentiate between the original and transformed versions of the text. Through extensive analysis, the authors demonstrated that the proposed framework achieved significant improvements in text anomaly detection.

Self-supervised models have indeed achieved remarkable success in various domains. However, their effectiveness is often dependent on the

Table 4
Performance of self-supervised models on CIFAR-10 against shallow and deep baselines. The bold values denote the highest AUROC (%) result for each class.

Class	Baseline					Self-predictive method							Contrastive learning			
	KDE	OCSVM	DSVDD	OCGAN	DROCC	GEOM	RotNet	OE	GOAD	Puzzle	SSLOE	PANDA	CSI	SSD	NDA	MSCL
Plane	61.2	65.6	61.7	75.7	81.7	74.7	71.9	87.6	77.2	78.9	90.4	97.4	89.9	82.7	**98.5**	97.7
Car	64.0	40.9	65.9	53.1	76.7	95.7	94.5	93.9	96.7	78.2	99.3	98.4	**99.1**	98.5	76.5	98.9
Bird	50.1	65.3	50.8	64.0	66.7	78.1	78.4	78.6	83.3	69.9	93.7	93.9	93.1	84.2	79.6	**95.8**
Cat	56.4	50.1	59.1	62.0	67.1	72.4	70.0	79.9	77.7	54.9	88.1	90.6	86.4	84.5	79.1	**94.5**
Deer	66.2	75.2	60.9	72.3	73.6	87.8	77.2	81.7	87.8	75.5	97.4	**97.5**	93.9	84.8	92.4	97.3
Dog	62.4	51.2	65.7	62.0	74.4	87.8	86.8	85.6	87.8	66.0	94.3	94.4	93.2	90.9	71.7	**97.1**
Frog	74.9	71.8	67.7	72.3	74.4	83.4	81.6	93.3	90.0	74.8	97.1	97.5	95.1	91.7	97.5	**98.4**
Horse	62.6	51.2	67.3	57.5	71.4	95.5	93.7	87.9	96.1	73.3	98.8	97.5	**98.7**	95.2	69.1	98.3
Ship	75.1	67.9	75.9	82.0	80.0	93.3	90.7	92.6	93.8	83.3	**98.7**	97.6	97.9	92.9	98.5	**98.7**
Truck	76.0	48.5	73.1	55.4	76.2	91.3	88.8	92.1	92.0	70.0	**98.5**	97.4	95.5	94.4	75.2	98.4
Ave:	64.8	58.8	64.8	65.7	74.2	86.0	83.3	87.3	88.2	72.5	95.6	96.2	94.3	90.0	84.3	**97.5**

specific transformations they employ, which can limit their applicability. Fortunately, certain transformations, such as data masking, have proven to be adaptable across different data types. Drawing inspiration from this observation, a dedicated line of research has emerged with the goal of developing self-supervised methods for anomaly detection that can be applied to diverse data types. This research aims to create techniques that leverage self-supervision to detect anomalies effectively and efficiently in various domains, expanding the scope of self-supervised anomaly detection beyond specific data types. The work of Qiu et al. (2021) is one of the most notable papers in this field. They have introduced the concept of trainable transformations that can be flexibly applied to any data type. The fundamental principle behind their approach involves mapping transformed data into a representation where distinct transformations can be discerned while still preserving the similarity between the transformed and original data. Remarkably, their framework demonstrates the capability to learn domain-specific transformations when applied to diverse datasets, including medical data and cyber-security data. This ability to adapt to different data types underscores the versatility and potential of their method in anomaly detection applications.

In conclusion, the field of self-supervised anomaly detection has expanded beyond image data, with significant progress made in detecting anomalies in non-image data types. By leveraging self-supervised learning techniques tailored to specific data modalities, researchers have demonstrated promising results in detecting anomalies in text, audio, time series, graphs, and IoT sensor data. These advancements open up new possibilities for anomaly detection in a wide range of applications, contributing to the development of robust and versatile anomaly detection systems.

8. Comparative evaluation and discussions

In this section, we focus on presenting the results reported by self-supervised image anomaly detection papers in a comparative manner to gain valuable insights into their performance. It is important to note that we have chosen to analyze only image data in this section, excluding other data types. This decision was made due to the inherent variations in datasets and backbones used across different studies, which could potentially introduce unfair comparisons. By focusing specifically on image data, we can provide a more meaningful and unbiased evaluation of the self-supervised anomaly detection methods.

A flurry of datasets is used to benchmark the self-supervised anomaly detection algorithms. CIFAR-10 (Krizhevsky, Nair, and Hinton), and MVTecAD (Bergmann et al., 2019) are two of the most common dataset that recent anomaly detection papers used. CIFAR-10 includes images of ten different objects. To benchmark an AD algorithm on this dataset, we assume that we only have access to the data from one of the classes during the training. During the test time, other classes are considered to be anomalies.

Table 4 presents the result of several state-of-the-art SSL models against the commonly used shallow and deep baselines for one-class AD on the CIFAR-10 dataset. This task can evaluate the performance of algorithms in semantic (high-level) anomaly detection. It is important to note that for the sake of fair comparison, we included the methods that use the same backbone. Looking at this table, we can readily confirm that the self-supervised approaches can outperform other shallow and deep anomaly detection algorithms by a significant margin. This remarkable improvement led to the emergence of SSL algorithms as a key category of anomaly detection.

Additionally, in Table 5, we present the outcomes achieved by the state-of-the-art self-supervised anomaly detection approaches when applied to the CIFAR-100 dataset. Notably, CIFAR-100 poses a greater challenge compared to CIFAR-10, primarily due to its increased number of classes (Reiss & Hoshen, 2021). The dataset's complexity is manifested in the diversity of objects and scenes across its extensive set of categories.

The results underscore the remarkable performance of Self-Supervised Learning (SSL) methods in tackling the complexities of the CIFAR-100 dataset. The superiority of SSL becomes particularly evident when confronted with the heightened difficulty posed by the dataset's expanded class structure. These methods showcase their capacity to discern anomalies in a more challenging environment, where traditional supervised approaches might encounter limitations. The CIFAR-100 dataset, with its broader spectrum of classes, serves as a robust benchmark to evaluate the robustness and adaptability of self-supervised anomaly detection techniques. The findings in Table 5 not only attest to the effectiveness of SSL methods but also highlight their potential for real-world applications where diverse and complex datasets are prevalent. This assertion is supported by the findings presented in Table 6, showcasing the average performance of several state-of-the-art self-supervised learning (SSL) methods on the challenging Imagenet dataset. Notably, even simpler SSL methods like RotNet demonstrate impressive performance, while more sophisticated approaches such as CLIP (Liznerski et al., 2022) exhibit excellent results.

Besides semantic anomaly detection, self-supervised methods show satisfactory performance for defect detection and spotting sensory anomalies (Kim et al., 2022; Song, Kong, Park, Kim, & Kang, 2021; Tsai et al., 2022). Fig. 6 shows the performance of the self-supervised models on the MVTecAD dataset against other widely-used algorithms including shallow models, deep models and generative models. More specifically, the compared shallow models are Gaussian (Ruff et al., 2021), MVE (Ruff et al., 2021), SVDD (Tax & Duin, 2004), KDE (Ruff et al., 2021), kPCA (Ruff et al., 2021), patch-SVDD (Yi & Yoon, 2020) and IGD (Chen, Tian, Pang, & Carneiro, 2021). The compared deep models are CAVGA (Venkataramanan, Peng, Singh, & Mahalanobis, 2020), ARNet (Fei et al., 2020), SPADE (Cohen & Hoshen, 2020), MOCCA (Valerio Massoli et al., 2020), DSVDD (Ruff et al., 2018), FCDD (Liznerski et al., 2020), DFR (Shi, Yang, & Qi, 2021), STFPM (Wang, Han, Ding, & Huang, 2021), Gaussian-AD (Rippel, Mertens, & Merhof, 2021), InTra (Pirnay & Chai, 2021), PaDiM (Defard, Setkov, Loesch, & Audigier, 2021) and DREAM (Zavrtanik, Kristan, & Skočaj, 2021). The included generative models in Fig. 6

Table 5
Performance of self-supervised models on CIFAR-100 against shallow and deep baselines. The bold values denote the highest AUROC (%) result for each class.

Class	DSVDD	DROC	ICAE	GEOM	Color	Count	Jigsaw	RotNet	CSI	PANDA	MSCL
0	57.4	82.9	66.0	74.7	70.1	79.2	80.4	82.8	86.3	91.5	**96.0**
1	63.0	84.3	60.1	68.5	54.2	71.1	73.3	75.2	84.8	92.6	**95.3**
2	70.0	88.6	59.2	74.0	68.7	75.3	75.6	77.4	88.9	**98.3**	98.1
3	55.8	86.4	58.7	81.0	65.3	81.6	80.2	85.6	85.7	96.6	**97.9**
4	69.0	92.6	60.9	78.4	62.1	76.4	78.9	80.1	93.7	96.3	**97.6**
5	51.0	84.5	54.2	59.1	51.1	66.5	64.3	67.4	81.9	94.1	**96.8**
6	59.9	73.4	63.7	81.8	75.4	82.9	84.2	87.1	91.8	96.4	**98.5**
7	53.0	84.2	66.1	65.0	61.9	66.4	68.2	66.3	83.9	91.2	**93.4**
8	51.6	87.7	74.8	85.5	75.5	87.5	86.3	89.4	91.6	94.7	**97.2**
9	72.9	94.1	78.3	90.6	72.1	86.9	89.1	90.8	95.0	94.0	**96.2**
10	81.5	85.2	80.4	87.6	68.3	86.2	88.2	88.3	94.0	96.4	**97.1**
11	53.6	87.8	69.3	83.9	74.2	81.1	84.6	85.2	90.1	92.6	**96.4**
12	50.6	82.0	75.6	83.2	66.5	77.5	79.2	80.1	90.3	93.1	**95.8**
13	44.0	82.7	61.0	58.0	53.2	56.3	58.1	60.3	81.5	89.4	**92.6**
14	57.2	93.4	64.3	92.1	78.4	90.7	92.9	94.9	94.4	98.0	**99.0**
15	47.7	75.8	66.3	68.3	62.1	69.9	70.4	73.6	85.6	89.7	**92.5**
16	54.3	80.3	72.0	73.5	57.8	73.2	74.8	76.4	83.0	92.1	**95.2**
17	74.7	97.5	75.9	93.8	70.4	96.3	96.0	97.8	97.5	97.7	**98.4**
18	52.1	94.4	67.4	90.7	71.1	89.4	91.5	92.1	95.9	94.7	**97.6**
19	57.9	92.4	64.8	85.0	76.2	85.7	86.3	90.6	95.2	92.7	**97.0**
Ave:	58.9	86.5	67.0	78.7	66.7	79.0	80.1	82.1	89.6	94.1	**96.4**

Table 6
Performance of self-supervised models on ImageNet. The bold values denote the highest AUROC (%).

RotNet	CSI	Count	Colorization	Jigsaw puzzle	CLIP
77.9	91.6	74.1	66.3	76.7	99.8

Table 7
Performance of self-supervised models on video datasets. The bold values denote the highest AUROC (%).

Dataset	Video colorization	Tracking	Shuffle and learn	AoT	RotNet	Sorting
UCF101	66.3	56.4	67.5	54.1	72.8	**74.6**
ILSVR2015	69.2	70.1	70.7	61.0	**74.4**	73.8

are AnoGAN (Schlegl et al., 2017), LSA (Abati, Porrello, Calderara, & Cucchiara, 2019), GANomaly (Akcay, Atapour-Abarghouei, & Breckon, 2018), AGAN (Ruff et al., 2021), Normalizing Flows-based DifferNet (Rudolph, Wandt, & Rosenhahn, 2021), CFLOW (Gudovskiy, Ishizaka, & Kozuka, 2022) and CS-Flow (Rudolph, Wehrbein, Rosenhahn, & Wandt, 2022). Looking at the figure, we can infer that SSL-based models can achieve a good performance on this dataset. However, the superiority of self-supervised algorithms over other baselines is less evident in this task than in one-class AD. Also, some algorithms such as GEOM, and CSI, which show state-of-the-art performance on CIFAR-10, achieve a weak accuracy in this anomaly detection task.

The above argument manifests the importance of choosing the right pretext task in self-supervised learning. Methods such as GEOM and RotNet which are based on geometric transformations, and CSI and SSD which are based on contrastive methods, work well for detecting semantic anomalies, but they are not well-suited for defect detection. On the other hand, SSL approaches that are based on pixel-level transformations, such as CutPaste, can achieve good accuracy on the MVTecAD dataset. Choosing the right proxy task, depending on the downstream objective and types of anomalies, is the key to the success of the SSL models. This allows researchers to improve the state-of-the-art by coming up with effective pretext tasks.

Expanding beyond the realm of anomaly detection in images, we conducted a comparative analysis of various self-supervised video anomaly detection methods, as shown in Table 7. This experiment encompasses two widely recognized benchmark datasets, namely UCF101 (Soomro, Zamir, & Shah, 2012) and ILSVR2015 (Russakovsky et al., 2015). The methods subjected to comparison include Video Colorization (Vondrick, Shrivastava, Fathi, Guadarrama, & Murphy, 2018), Tracking (Wang & Gupta, 2015), Shuffle and Learn (Misra, Zitnick, & Hebert, 2016), AoT (Wei, Lim, Zisserman, & Freeman, 2018), RotNet (Gidaris et al., 2018), and Sorting (Lee, Huang, Singh, & Yang, 2017).

The findings in the table unveil a consistent trend across all methods, showcasing their ability to surpass chance-level performance. In the domain of video anomaly detection, self-supervision can be used for acquiring efficient feature representations in both temporal and spatial dimensions. Notably, Tracking and AoT heavily rely on temporal features, while other methods prioritize the acquisition of robust spatial features (Ali, Khan, & Kyung, 2020).

An important pattern emerges from the comparative results: spatial features tend to contribute more meaningfully to the task of visual anomaly detection in videos compared to their temporal counterparts. While temporal information remains crucial, the emphasis on spatial features among several methods underscores the significance of capturing contextual and structural intricacies within video data. This nuanced understanding provides valuable insights for the design and optimization of self-supervised video anomaly detection models, emphasizing the interplay between temporal and spatial feature learning for enhanced performance.

Out-of-distribution detection is another task in which SSL models are widely applied. Table 8 shows the experimental results of some SSL models (shown in the top 10 rows) against a supervised method, shown in the last row of the table. The supervised method is in fact a ResNet-50 network that is trained to classify the data available in CIFAR-10 from the other OOD dataset – i.e., ResNet-50 is trained as an eleven-way classifier, ten for CIFAR-10 and one for the OOD dataset. To benchmark an OOD algorithm, it is common to train a model on the CIFAR-10 dataset and test the model using another dataset. If the samples of the test datasets are similar to the CIFAR-10 to some extent, the task is called near-OOD detection (e.g. CIFAR-10 vs. CIFAR-100).

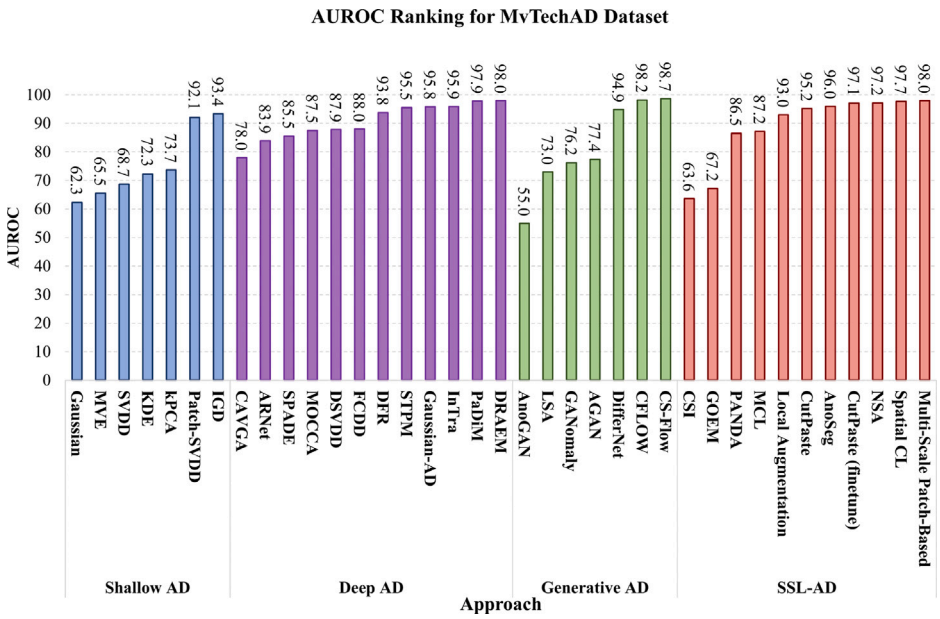


Fig. 6. Performance of anomaly detection algorithms on the MVTecAD dataset. Each group of algorithm is denoted by a different color.

Table 8
Performance of SSL models against a supervise-based method for OOD detection. The bold values denote the highest AUROC (%) result for each OOD dataset.

Method	IND: CIFAR-10			IND: CIFAR-100		
	OOD: CIFAR-100	SVHN	LSUN	OOD: CIFAR-10	SVHN	LSUN
RotNet (Hendrycks et al., 2018)	93.3	94.4	**97.6**	75.7	86.9	**93.4**
CSI (Tack et al., 2020)	89.2	99.8	90.3	–	–	–
SSL-OE (Hendrycks et al., 2019)	93.3	98.4	93.2	–	–	–
CLP (Winkens et al., 2020)	92.9	99.5	–	**78.9**	95.4	–
SSL-OOD (Mohseni et al., 2020)	93.8	99.2	98.9	77.7	95.8	88.9
MCL (Cho, Seol, & Lee, 2021)	90.8	97.9	93.8	–	–	–
MCL-SEI (Cho, Seol, & Lee, 2021)	94.0	99.3	96.3	–	–	–
SSD (Sehwag et al., 2021)	90.6	99.6	96.5	69.6	94.9	79.5
SSD_k (k=5) (Sehwag et al., 2021)	93.1	99.7	97.8	78.3	**99.1**	93.4
SDNS (Rafiee et al., 2022)	**94.2**	**99.9**	97.5	67.6	97.2	74.6
ResNet-50 (Sehwag et al., 2021)	90.6	99.6	93.8	55.3	94.5	69.4

Otherwise, it is referred to as far-OOD detection (e.g. CIFAR-10 vs. SVHN, or CIFAR-10 vs. LSUN). We observe that SSL can even achieve better performance than the supervised baseline. This manifests that it is not necessary to have access to the data ground truths for the OOD detection task.

The results reported in Table 8 shows that all the SSL-based methods can achieve an accuracy above 94% on far-OOD detection (i.e. CIFAR-10 vs. SVHN). It can suggest that the SSL models can learn meaningful features of the dataset. Almost all algorithms perform well in near-OOD detection, and some can even beat the supervised baseline.

9. Application domains

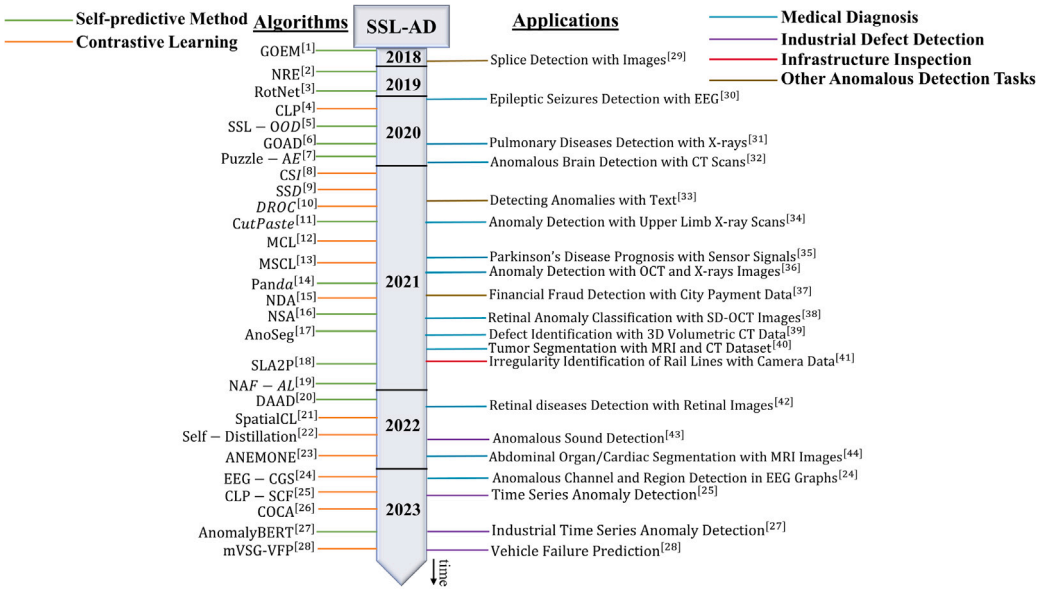
Anomaly detection systems are widely deployed in various domains, such as medicine, industry, infrastructure, social medical, financial security, etc. Despite the fact that self-supervised anomaly detection is a relatively new field, it is now widely employed in practical applications along with other popular methods such as Semi-supervised learning (Villa-Perez et al., 2021), and GAN and its variants (Xia et al., 2022).

Self-supervised learning algorithms are commonly used in medical research for detecting irregularities in patients' records. They are successfully employed for detecting epileptic seizures (Xu, Zheng, Mao, Wang, & Zheng, 2020), pulmonary diseases (Bozorgtabar, Mahapatra, Vray, & Thiran, 2020), Parkinson disease (Jiang et al., 2021), retinal diseases (Burlina, Paul, Liu, & Bressler, 2022), and heart disorders (Ho & Armanfard, 2023a). In addition, they are applied to different modalities of medical data, including Computed Tomography (CT) scans (Venkatakrishnan, Kim, Eisawy, Pfister, & Navab, 2020), 3D volumetric CT data (Cho, Kang, & Park, 2021), X-ray scans (Spahr, Bozorgtabar, & Thiran, 2021), optical coherence tomography (OCT) (Zhao et al., 2021), Spectral Domain - optical coherence tomography images (SD-OCT) (Park, Balint, & Hwang, 2021), MRI images (Hansen, Gautam, Jenssen, & Kampffmeyer, 2022; Zhang, Xie, Huang, Zhang, & Wang, 2021), videos and physiological signals (Ho & Armanfard, 2023b).

Self-supervised anomaly detection method are also employed in industrial applications for defect detection, and failure prediction (Bahavan et al., 2020; Hou, Tao, & Xu, 2021), as well as for monitoring infrastructural facilities (Jahan, Umesh, & Roth, 2021; Liu, Xu, & Xu, 2021).

The application of self-supervised AD is not limited to the aforementioned areas. Several fields such as financial fraud detection (Schreyer, Sattarov, & Borth, 2021; Wang, Dou, et al., 2021), text anomaly detection (Manolache et al., 2021b), and splice detection (Huh, Liu, Owens, & Efros, 2018) are also benefited from the SSL algorithms.

Fig. 7 depicts the timeline of papers focusing on self-supervised anomaly detection algorithms and their applications. This figure highlights the rapid growth of this field and its wide applicability in addressing real-world problems.



[1](Golan and El-Yaniv, 2018); [2](Sabokrou et al., 2019); [3](Hendrycks et al., 2018); [4](Winkens et al., 2020); [5](Mohseni et al., 2020); [6](Bergman and Hoshen, 2020); [7](Salehi et al., 2020); [8](Tack et al., 2020); [9](Sehwag et al., 2021); [10](Sohn et al., 2020); [11](Li et al., 2021); [12](Cho et al., 2021a); [13](Reiss and Hoshen, 2021); [14](Reiss et al., 2021); [15](Chen et al., 2021a); [16](Schlüter et al., 2021); [17](Song et al., 2021); [18](Wang et al., 2021c); [19](Zhang et al., 2021a); [20](Zhang et al., 2022a); [21](Kim et al., 2022); [22](Rafiee et al., 2022); [23](Zheng et al., 2022); [24](Ho and Armanfard, 2023b); [25](Wang et al., 2023); [26](Guan et al., 2023); [27](Jeong et al., 2023); [28](Hojjati et al., 2023); [29](Huh et al., 2018); [30](Xu et al., 2020); [31](Bozorgtabar et al., 2020); [32](Venkatakrishnan et al., 2020); [33](Manolache et al., 2021a); [34](Spahr et al., 2021); [35](Jiang et al., 2021); [36](Zhao et al., 2021); [37](Schreyer et al., 2021); [38](Park et al., 2021); [39](Cho et al., 2021b); [40](Zhang et al., 2021b); [41](Jahan et al., 2021); [42](Burlina et al., 2022); [43](Bai et al., 2023); [44](Hansen et al., 2022).

Fig. 7. Timeline of Self-Supervised Anomaly Detection Papers. Papers concerning the algorithms are distinguished from the application papers. The category of each algorithm is denoted by a distinctive color.

10. Future directions

Although the self-supervised models have established themselves as state-of-the-art in anomaly detection, there is still much room for improvement in this research field. This section briefly discusses some critical challenges that SSL-based anomaly detectors suffer from and presents some high-level ideas for addressing them.

10.1. Negative sampling in contrastive models

In recent years, contrastive models dominated self-supervised AD algorithms. To learn an efficient representation, CL algorithms require accessing negative samples. In the standard setting, it is assumed that other batch samples are negative, even though their class label is the same as that of the query sample. However, if the number of same-class samples increases, the quality of the learned representation degrades. In some anomaly detection tasks, where the training data comprises samples of one class, this negative sampling bias may turn into a big issue. This motivates researchers to design unbiased versions of the contrastive loss (Chuang, Robinson, Lin, Torralba, & Jegelka, 2020).

Interestingly, previous studies showed that even in the one-class setting, the instance discrimination contrastive learning can lead to a suitable representation for anomalies. This can be because all the training data are spread out on a hypersphere, and the anomalies are mapped to the center of the space, as we discussed in Section 6.

Following the success of SimCLR, several other contrastive models are developed. These methods can be good candidates for one-class anomaly detection since they can be trained using only positive samples. Some recent models, such as BYOL (Grill et al., 2020) and Barlow Twins (Zbontar, Jing, Misra, LeCun, & Deny, 2021), do not require negative samples during training. To the best of our knowledge, there is no study that evaluates the performance of these models for anomaly detection.

10.2. Incorporating labeled data

In the most anomaly detection studies, it is assumed that no labeled anomaly is available during the training phase. However, in some applications, we might be able to have a few labeled anomalies. These labeled samples can significantly improve the algorithm if incorporated appropriately. Recently, Sehwag et al. (2021) explored the problem of few-shot anomaly detection, where they assume a few labeled anomalies are present. They showed that even a few anomalies can significantly improve the detection accuracy. Zheng et al. (2022) proposed an extended algorithm of multi-scale contrastive learning, called ANEMONE, by incorporating it with a handful of ground-truth anomalies. Since the assumption of having access to a few anomaly samples during training time is feasible in many tasks, we believe that models with the capability to incorporate them have a great potential to improve the detection performance. Such methods also have more application in real-life problems.

10.2.1. Multi-modal anomaly detection

In many applications, including medical imaging, cybersecurity, and surveillance systems, the datasets contain multiple sources of information or modalities. Detecting anomalies in such cases heavily depends on the quality and relevance of the information contained in each modality and the ability to effectively fuse this information to make a

robust decision. Since self-supervised methods have already established themselves as powerful tools for learning representations, it would be interesting to study their application in multi-modal learning for anomaly detection. To this end, researchers might pursue the direction of designing cross-modal proxy tasks that aid the model to fuse information from different modalities in an efficient manner.

10.2.2. Efficient self-supervised learning

Currently, self-supervised models have shown superior performance over traditional algorithms. Yet, they face critical challenges such as their computational cost, which prevents their widespread use in many applications. Future research in self-supervised learning will likely focus on designing computationally effective models that can leverage the vast amounts of unannotated data available for training. Additionally, the use of transfer learning, meta-learning, and federated learning may become more widespread as a way to overcome the limitations of self-supervised algorithms and enable their deployment in resource-constrained environments. Furthermore, research may also investigate the scalability of self-supervised learning to handle large amounts of data and diverse domains, as well as its interpretability and robustness to adversarial attacks.

11. Conclusion

In this paper, we discussed the state-of-the-art methods in self-supervised anomaly detection and highlighted the strengths and drawbacks of each approach. We also compared their performance on benchmark datasets and pinpointed their applications. In summary, we can argue that self-supervised models are well suited for tackling the problem of anomaly detection. Yet, there are still a lot of under-explored issues and room for improvement. Still, the significant success of SSL algorithms offers a bright horizon for achieving new milestones in automatic anomaly detection.

CRediT authorship contribution statement

Hadi Hojjati: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Thi Kieu Khanh Ho:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Narges Armanfard:** Conceptualization, Funding acquisition, Project administration, Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Hadi Hojjati reports financial support was provided by Quebec Research Fund Nature and Technology. Narges Armanfard reports financial support was provided by Natural Sciences and Engineering Research Council of Canada. Hadi Hojjati reports financial support was provided by AGE-WELL NCE Inc.

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors wish to acknowledge the financial support of the Natural Sciences, Engineering Research Council of Canada (NSERC), Fonds de recherche du Québec (FRQNT), AGE-WELL, Canada, and the Department of Electrical and Computer Engineering at McGill University, Canada. This research was enabled in part by support provided by Calcul Quebec and Compute Canada.

References

Abati, D., Porrello, A., Calderara, S., & Cucchiara, R. (2019). Latent space autoregression for novelty detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 481–490).

Agyemang, M., Barker, K., & Alhajj, R. (2006). A comprehensive survey of numeric and symbolic outlier mining techniques. *Intelligent Data Analysis*, *10*, 521–538. http://dx.doi.org/10.3233/IDA-2006-10604.

Akcay, S., Atapour-Abarghouei, A., & Breckon, T. P. (2018). Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision* (pp. 622–637). Springer.

Ali, R., Khan, M. U. K., & Kyung, C. M. (2020). Self-supervised representation learning for visual anomaly detection. arXiv:2006.09654.

Azizi, S., Mustafa, B., Ryan, F., Beaver, Z., Freyberg, J., Deaton, J., et al. (2021). Big self-supervised models advance medical image classification. In *2021 IEEE/CVF international conference on computer vision* (pp. 3458–3468). http://dx.doi.org/10.1109/ICCV48922.2021.00346.

Bahavan, N., Suman, N., Cader, S., Ranganayake, R., Seneviratne, D., Maddumage, V., et al. (2020). Anomaly detection using deep reconstruction and forecasting for autonomous systems. arXiv preprint arXiv:2006.14556.

Bai, J., Chen, J., Wang, M., Ayub, M. S., & Yan, Q. (2023). SSDPT: Self-supervised dual-path transformer for anomalous sound detection. *Digital Signal Processing*, *135*, Article 103939. http://dx.doi.org/10.1016/j.dsp.2023.103939, URL: https://www.sciencedirect.com/science/article/pii/S1051200423000349.

Bergman, L., & Hoshen, Y. (2020). Classification-based anomaly detection for general data. In *International conference on learning representations*.

Bergmann, P., Fauser, M., Sattlegger, D., & Steger, C. (2019). MVTec AD — A comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF conference on computer vision and pattern recognition* (pp. 9584–9592). http://dx.doi.org/10.1109/CVPR.2019.00982.

Bozorgtabar, B., Mahapatra, D., Vray, G., & Thiran, J. P. (2020). SALAD: Self-supervised aggregation learning for anomaly detection on x-rays. In *International conference on medical image computing and computer-assisted intervention* (pp. 468–478). Springer.

Burlina, P., Paul, W., Liu, T. A., & Bressler, N. M. (2022). Detecting anomalies in retinal diseases using generative, discriminative, and self-supervised deep learning. *JAMA Ophthalmology*, *140*(2), 185–189.

Carmona, C. U., Aubet, F. X., Flunkert, V., & Gasthaus, J. (2022). Neural contextual anomaly detection for time series. In L. D. Raedt (Ed.), *Proceedings of the thirty-first international joint conference on artificial intelligence* (pp. 2843–2851). International Joint Conferences on Artificial Intelligence Organization, http://dx.doi.org/10.24963/ijcai.2022/394, Main Track.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., & Joulin, A. (2020). Unsupervised learning of visual features by contrasting cluster assignments. In *Proceedings of advances in neural information processing systems*.

Chalapathy, R., & Chawla, S. (2019). Deep learning for anomaly detection: A survey. http://dx.doi.org/10.48550/ARXIV.1901.03407, URL: https://arxiv.org/abs/1901.03407.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Outlier detection: A survey. *ACM Computing Surveys*, *14*, 15.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607). PMLR.

Chen, Y., Tian, Y., Pang, G., & Carneiro, G. (2021). Unsupervised anomaly detection with multi-scale interpolated gaussian descriptors, 2. arXiv preprint arXiv:2101.10043.

Chen, C., Xie, Y., Lin, S., Qiao, R., Zhou, J., Tan, X., et al. (2021). Novelty detection via contrastive learning with negative data augmentation. In Z. H. Zhou (Ed.), *Proceedings of the thirtieth international joint conference on artificial intelligence* (pp. 606–614). International Joint Conferences on Artificial Intelligence Organization, http://dx.doi.org/10.24963/ijcai.2021/84, Main Track.

Chen, Z., Yeo, C. K., Lee, B. S., & Lau, C. T. (2018). Autoencoder-based network anomaly detection. In *2018 Wireless telecommunications symposium* (pp. 1–5). http://dx.doi.org/10.1109/WTS.2018.8363930.

Chen, B., Zhang, J., Zhang, X., Dong, Y., Song, J., Zhang, P., et al. (2022). GCCAD: Graph contrastive learning for anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*.

Cho, J., Kang, I., & Park, J. (2021). Self-supervised 3D out-of-distribution detection via pseudoanomaly generation. In *International conference on medical image computing and computer-assisted intervention* (pp. 95–103). Springer.

Cho, H., Seol, J., & Lee, S.-g. (2021). Masked contrastive learning for anomaly detection. In Z. H. Zhou (Ed.), *Proceedings of the thirtieth international joint conference on artificial intelligence* (pp. 1434–1441). International Joint Conferences on Artificial Intelligence Organization, http://dx.doi.org/10.24963/ijcai.2021/198, Main Track.

Chopra, S., Hadsell, R., & LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE computer society conference on computer vision and pattern recognition, vol. 1*. http://dx.doi.org/10.1109/CVPR.2005.202.

Chuang, C. Y., Robinson, J., Lin, Y. C., Torralba, A., & Jegelka, S. (2020). Debiased contrastive learning. *Advances in Neural Information Processing Systems*, *33*.

Cohen, N., & Hoshen, Y. (2020). Sub-image anomaly detection with deep pyramid correspondences. arXiv preprint arXiv:2005.02357.

Defard, T., Setkov, A., Loesch, A., & Audigier, R. (2021). Padim: a patch distribution modeling framework for anomaly detection and localization. In *International conference on pattern recognition* (pp. 475–489). Springer.

Di Mattia, F., Galeone, P., De Simoni, M., & Ghelfi, E. (2019). A survey on GANs for anomaly detection. http://dx.doi.org/10.48550/ARXIV.1906.11632, URL: https://arxiv.org/abs/1906.11632.

Doersch, C., Gupta, A., & Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision* (pp. 1422–1430).

Duan, J., Wang, S., Zhang, P., Zhu, E., Hu, J., Jin, H., et al. (2022). Graph anomaly detection via multi-scale contrastive learning networks with augmented view. arXiv:2212.00535.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. http://dx.doi.org/10.1016/j.patrec.2005.10.010.

Fei, Y., Huang, C., Jinkun, C., Li, M., Zhang, Y., & Lu, C. (2020). Attribute restoration framework for anomaly detection. *IEEE Transactions on Multimedia*.

Feinman, R., Curtin, R. R., Shintre, S., & Gardner, A. B. (2017). Detecting adversarial samples from artifacts. arXiv preprint arXiv:1703.00410.

Field, S. A., Tyre, A. J., Jonzén, N., Rhodes, J. R., & Possingham, H. P. (2004). Minimizing the cost of environmental management decisions by optimizing statistical thresholds. *Ecology Letters*, *7*(8), 669–675.

Fu, Y., & Xue, F. (2022). MAD: Self-supervised masked anomaly detection task for multivariate time series. In *2022 International joint conference on neural networks* (pp. 1–8). http://dx.doi.org/10.1109/IJCNN55064.2022.9892218.

Gidaris, S., Singh, P., & Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728.

Giri, R., Tenneti, S. V., Cheng, F., Helwani, K., Isik, U., & Krishnaswamy, A. (2020). Self-supervised classification for detecting anomalous sounds. In *Detection and classification of acoustic scenes and events workshop 2020*. URL: https://www.amazon.science/publications/self-supervised-classification-for-detecting-anomalous-sounds.

Golan, I., & El-Yaniv, R. (2018). Deep anomaly detection using geometric transformations. *Advances in Neural Information Processing Systems*, *31*.

Grill, J. B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., et al. (2020). Bootstrap your own latent: A new approach to self-supervised learning. arXiv:2006.07733.

Guan, J., Xiao, F., Liu, Y., Zhu, Q., & Wang, W. (2023). Anomalous sound detection using audio representation with machine ID based contrastive learning pretraining. In *ICASSP 2023 - 2023 IEEE international conference on acoustics, speech and signal processing* (pp. 1–5). http://dx.doi.org/10.1109/ICASSP49357.2023.10096054.

Gudovskiy, D., Ishizaka, S., & Kozuka, K. (2022). Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 98–107).

Gui, J., Chen, T., Zhang, J., Cao, Q., Sun, Z., Luo, H., et al. (2023). A survey of self-supervised learning from multiple perspectives: Algorithms, applications and future trends. arXiv:2301.05712.

Hansen, S., Gautam, S., Jenssen, R., & Kampffmeyer, M. (2022). Anomaly detection-inspired few-shot medical image segmentation through self-supervision with supervoxels. *Medical Image Analysis*, *78*, Article 102385.

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9729–9738).

Hendrycks, D., Mazeika, M., & Dietterich, T. (2018). Deep anomaly detection with outlier exposure. In *International conference on learning representations*. URL: https://openreview.net/forum?id=HyxCxhRcY7.

Hendrycks, D., Mazeika, M., Kadavath, S., & Song, D. (2019). Using self-supervised learning can improve model robustness and uncertainty. *Advances in Neural Information Processing Systems*, *32*.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., et al. (2019). Learning deep representations by mutual information estimation and maximization. In *International conference on learning representations*. URL: https://openreview.net/forum?id=Bklr3j0cKX.

Ho, T. K. K., & Armanfard, N. (2023a). Multivariate time-series anomaly detection with contaminated data: Application to physiological signals. arXiv preprint arXiv:2308.12563.

Ho, T. K. K., & Armanfard, N. (2023b). Self-supervised learning for anomalous channel detection in EEG graphs: Application to seizure analysis. In *Proceedings of the AAAI conference on artificial intelligence*.

Ho, T. K. K., Karami, A., & Armanfard, N. (2023). Graph-based time-series anomaly detection: A survey. arXiv preprint arXiv:2302.00058.

Hodge, V., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, *22*, 85–126. http://dx.doi.org/10.1023/B:AIRE.0000045502.10941.a9.

Hojjati, H., & Armanfard, N. (2022). Self-supervised acoustic anomaly detection via contrastive learning. In *ICASSP 2022 - 2022 IEEE international conference on acoustics, speech and signal processing*.

Hojjati, H., & Armanfard, N. (2023). DASVDD: deep autoencoding support vector data descriptor for anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 1–12. http://dx.doi.org/10.1109/TKDE.2023.3328882.

Hojjati, H., Sadeghi, M., & Armanfard, N. (2023). Multivariate time-series anomaly detection with temporal self-supervision and graphs: Application to vehicle failure prediction. In *The European conference on machine learning and principles and practice of knowledge discovery in databases*.

Hou, W., Tao, X., & Xu, D. (2021). A self-supervised CNN for particle inspection on optical element. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1–12.

Huang, T., Pei, Y., Menkovski, V., & Pechenizkiy, M. (2022). Hop-count based self-supervised anomaly detection on attributed networks. In *Joint European conference on machine learning and knowledge discovery in databases* (pp. 225–241). Springer.

Huang, D., Shen, L., Yu, Z., Zheng, Z., Huang, M., & Ma, Q. (2022). Efficient time series anomaly detection by multiresolution self-supervised discriminative network. *Neurocomputing*, *491*, 261–272. http://dx.doi.org/10.1016/j.neucom.2022.03.048, URL: https://www.sciencedirect.com/science/article/pii/S0925231222003435.

Huh, M., Liu, A., Owens, A., & Efros, A. A. (2018). Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European conference on computer vision* (pp. 101–117).

Jahan, K., Umesh, J. P., & Roth, M. (2021). Anomaly detection on the rail lines using semantic segmentation and self-supervised learning. In *2021 IEEE symposium series on computational intelligence* (pp. 1–7). IEEE.

Jeong, Y., Yang, E., Ryu, J. H., Park, I., & Kang, M. (2023). AnomalyBERT: Self-supervised transformer for time series anomaly detection using data degradation scheme. arXiv:2305.04468.

Jiang, H., Lim, W. Y. B., Ng, J. S., Wang, Y., Chi, Y., & Miao, C. (2021). Towards Parkinson's disease prognosis using self-supervised learning and anomaly detection. In *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing* (pp. 3960–3964). IEEE.

Jiao, Y., Yang, K., Song, D., & Tao, D. (2022). TimeAutoAD: Autonomous anomaly detection with self-supervised contrastive loss for multivariate time series. *IEEE Transactions on Network Science and Engineering*, *9*(3), 1604–1619. http://dx.doi.org/10.1109/TNSE.2022.3148276.

Jing, L., & Tian, Y. (2021). Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(11), 4037–4058. http://dx.doi.org/10.1109/TPAMI.2020.2992393.

Jumutc, V., & Suykens, J. A. (2014). Multi-class supervised novelty detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(12), 2510–2523.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., et al. (2020). Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), *Advances in neural information processing systems, vol. 33* (pp. 18661–18673). Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

Kim, S., Choi, Y., & Lee, M. (2015). Deep learning with support vector data description. *Neurocomputing*, *165*, 111–117.

Kim, M., Ho, M. T., & Kang, H. G. (2021). Self-supervised complex network for machine sound anomaly detection. In *2021 29th European signal processing conference* (pp. 586–590). http://dx.doi.org/10.23919/EUSIPCO54536.2021.9615923.

Kim, D., Jeong, D., Kim, H., Chong, K., Kim, S., & Cho, H. (2022). Spatial contrastive learning for anomaly detection and localization. *IEEE Access*, *10*, 17366–17376.

Kiran, B. R., Thomas, D. M., & Parakkal, R. (2018). An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *Journal of Imaging*, *4*(2), 36.

Krizhevsky, A., Nair, V., & Hinton, G. (2009). CIFAR-10 (Canadian institute for advanced research). URL: http://www.cs.toronto.edu/~kriz/cifar.html.

Larsson, G., Maire, M., & Shakhnarovich, G. (2016). Learning representations for automatic colorization. In *European conference on computer vision*.

Latif, S., Usman, M., Rana, R., & Qadir, J. (2018). Phonocardiographic sensing using deep learning for abnormal heartbeat detection. *IEEE Sensors Journal*, *18*(22), 9393–9400.

Lee, H. Y., Huang, J. B., Singh, M., & Yang, M. H. (2017). Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE international conference on computer vision* (pp. 667–676).

Lee, K., Lee, K., Lee, H., & Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, *31*.

Li, C. L., Sohn, K., Yoon, J., & Pfister, T. (2021). Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9664–9674).

Liu, S., Garrepalli, R., Dietterich, T., Fern, A., & Hendrycks, D. (2018). Open category detection with PAC guarantees. In J. Dy, & A. Krause (Eds.), *Proceedings of machine learning research: vol. 80, Proceedings of the 35th international conference on machine learning* (pp. 3169–3178). PMLR, URL: https://proceedings.mlr.press/v80/liu18e.html.

Liu, Y., Jin, M., Pan, S., Zhou, C., Zheng, Y., Xia, F., et al. (2022). Graph self-supervised learning: A survey. *IEEE Transactions on Knowledge and Data Engineering*, *35*(6), 5879–5900.

Liu, Y., Li, Z., Pan, S., Gong, C., Zhou, C., & Karypis, G. (2021). Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(6), 2378–2392.

Liu, Y., Pan, S., Wang, Y. G., Xiong, F., Wang, L., Chen, Q., et al. (2021). Anomaly detection in dynamic graphs via transformer. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, M., Xu, Z., & Xu, Q. (2021). DeepFIB: Self-imputation for time series anomaly detection. arXiv preprint arXiv:2112.06247.

Liznerski, P., Ruff, L., Vandermeulen, R. A., Franks, B. J., Kloft, M., & Müller, K. R. (2020). Explainable deep one-class classification. arXiv preprint arXiv:2007.01760.

Liznerski, P., Ruff, L., Vandermeulen, R. A., Franks, B. J., Muller, K. R., & Kloft, M. (2022). Exposing outlier exposure: What can be learned from few, one, and zero outlier images. *Transactions on Machine Learning Research*, URL: https://openreview.net/forum?id=3v78awEzyB.

Luo, X., Wu, J., Yang, J., Xue, S., Peng, H., Zhou, C., et al. (2022). Deep graph level anomaly detection with contrastive learning. *Scientific Reports*, *12*(1), 19867.

Mahalanobis, P. (1936). On the generalised distance in statistics. In *Proceedings of the national institute of sciences of India, vol. 2, no. 1* (pp. 49–55).

Malaiya, R. K., Kwon, D., Kim, J., Suh, S. C., Kim, H., & Kim, I. (2018). An empirical evaluation of deep learning for network anomaly detection. In *2018 International conference on computing, networking and communications* (pp. 893–898). IEEE.

Manolache, A., Brad, F., & Burceanu, E. (2021a). DATE: Detecting anomalies in text via self-supervision of transformers. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: Human language technologies* (pp. 267–277). Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2021.naacl-main.25, Online, URL: https://aclanthology.org/2021.naacl-main.25.

Manolache, A., Brad, F., & Burceanu, E. (2021b). Date: Detecting anomalies in text via self-supervision of transformers. arXiv preprint arXiv:2104.05591.

Min, E., Long, J., Liu, Q., Cui, J., Cai, Z., & Ma, J. (2018). Su-ids: A semi-supervised and unsupervised framework for network intrusion detection. In *International conference on cloud computing and security* (pp. 322–334). Springer.

Misra, I., Zitnick, C. L., & Hebert, M. (2016). Shuffle and learn: Unsupervised learning using temporal order verification. In *European conference on computer vision* (pp. 527–544). Springer.

Mohseni, S., Pitale, M., Yadawa, J., & Wang, Z. (2020). Self-supervised learning for generalizable out-of-distribution detection. In *Proceedings of the AAAI conference on artificial intelligence, vol. 34, no. 04* (pp. 5216–5223). http://dx.doi.org/10.1609/aaai.v34i04.5966, URL: https://ojs.aaai.org/index.php/AAAI/article/view/5966.

Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. *ACM Computing Surveys*, *54*(2), http://dx.doi.org/10.1145/3439950.

Park, S., Balint, A., & Hwang, H. (2021). Self-supervised medical out-of-distribution using U-net vision transformers. In *International conference on medical image computing and computer-assisted intervention* (pp. 104–110). Springer.

Pirnay, J., & Chai, K. (2021). Inpainting transformer for anomaly detection. arXiv preprint arXiv:2104.13897.

Qiu, C., Pfrommer, T., Kloft, M., Mandt, S., & Rudolph, M. (2021). Neural transformation learning for deep anomaly detection beyond images. In M. Meila, & T. Zhang (Eds.), *Proceedings of machine learning research*: *vol. 139*, *Proceedings of the 38th international conference on machine learning* (pp. 8703–8714). PMLR, URL: https://proceedings.mlr.press/v139/qiu21a.html.

Rafiee, N., Gholamipoorfard, R., Adaloglou, N., Jaxy, S., Ramakers, J., & Kollmann, M. (2022). Self-supervised anomaly detection by self-distillation and negative sampling. arXiv preprint arXiv:2201.06378.

Ravanelli, M., Zhong, J., Pascual, S., Swietojanski, P., Monteiro, J., Trmal, J., et al. (2020). Multi-task self-supervised learning for robust speech recognition. In *ICASSP 2020 - 2020 IEEE international conference on acoustics, speech and signal processing* (pp. 6989–6993). http://dx.doi.org/10.1109/ICASSP40776.2020.9053569.

Reiss, T., Cohen, N., Bergman, L., & Hoshen, Y. (2021). Panda: Adapting pretrained features for anomaly detection and segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2806–2814).

Reiss, T., & Hoshen, Y. (2021). Mean-shifted contrastive loss for anomaly detection. arXiv preprint arXiv:2106.03844.

Rippel, O., Mertens, P., & Merhof, D. (2021). Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *2020 25th international conference on pattern recognition* (pp. 6726–6733). IEEE.

Rudolph, M., Wandt, B., & Rosenhahn, B. (2021). Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 1907–1916).

Rudolph, M., Wehrbein, T., Rosenhahn, B., & Wandt, B. (2022). Fully convolutional cross-scale-flows for image-based defect detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 1088–1097).

Ruff, L., Kauffmann, J. R., Vandermeulen, R. A., Montavon, G., Samek, W., Kloft, M., et al. (2021). A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., et al. (2018). Deep one-class classification. In *International conference on machine learning* (pp. 4393–4402). PMLR.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K. R., et al. (2019). Deep semi-supervised anomaly detection. arXiv preprint arXiv:1906.02694.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ILSVRC 2015: Object detection from video and object detection from images challenges. In *Proceedings of the IEEE international conference on computer vision* (pp. 3376–3385).

Sabokrou, M., Fayyaz, M., Fathy, M., Moayed, Z., & Klette, R. (2018). Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes. *Computer Vision and Image Understanding*, *172*, 88–97. http://dx.doi.org/10.1016/j.cviu.2018.02.006.

Sabokrou, M., Khalooei, M., & Adeli, E. (2019). Self-supervised representation learning via neighborhood-relational encoding. In *Proceedings of the IEEE/CVF international conference on computer vision*.

Sadeghi, M., Hojjati, H., & Armanfard, N. (2023). C3: Cross-instance guided contrastive clustering. *British Machine Vision Conference*.

Salehi, M., Eftekhar, A., Sadjadi, N., Rohban, M. H., & Rabiee, H. R. (2020). Puzzle-AE: Novelty detection in images through solving puzzles. arXiv:2008.12959.

Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G., & Schmidt-Erfurth, U. (2019). f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, *54*, 30–44.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., & Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging* (pp. 146–157). Springer.

Schlüter, H. M., Tan, J., Hou, B., & Kainz, B. (2021). Self-supervised out-of-distribution detection and localization with natural synthetic anomalies (NSA). arXiv preprint arXiv:2109.15222.

Schölkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J., & Platt, J. (1999). Support vector method for novelty detection. In *Proceedings of the 12th international conference on neural information processing systems* (pp. 582–588). Cambridge, MA, USA: MIT Press.

Schreyer, M., Sattarov, T., & Borth, D. (2021). Multi-view contrastive self-supervised learning of accounting data representations for downstream audit tasks. arXiv preprint arXiv:2109.11201.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE conference on computer vision and pattern recognition* (pp. 815–823). http://dx.doi.org/10.1109/CVPR.2015.7298682.

Sehwag, V., Chiang, M., & Mittal, P. (2021). {SSD}: A unified framework for self-supervised outlier detection. In *International conference on learning representations*. URL: https://openreview.net/forum?id=v5gjXpmR8J.

Shenkar, T., & Wolf, L. (2022). Anomaly detection for tabular data with internal contrastive learning. In *International conference on learning representations*.

Shi, Y., Yang, J., & Qi, Z. (2021). Unsupervised anomaly segmentation via deep feature reconstruction. *Neurocomputing*, *424*, 9–22.

Sohn, K., Li, C. L., Yoon, J., Jin, M., & Pfister, T. (2020). Learning and evaluating representations for deep one-class classification. arXiv preprint arXiv:2011.02578.

Song, J., Kong, K., Park, Y. I., Kim, S. G., & Kang, S. J. (2021). AnoSeg: Anomaly segmentation network using self-supervised learning. arXiv preprint arXiv:2110.03396.

Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv:1212.0402.

Spahr, A., Bozorgtabar, B., & Thiran, J. P. (2021). Self-taught semi-supervised anomaly detection on upper limb X-rays. In *2021 IEEE 18th international symposium on biomedical imaging* (pp. 1632–1636). IEEE.

Tack, J., Mo, S., Jeong, J., & Shin, J. (2020). Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in Neural Information Processing Systems*, *33*, 11839–11852.

Tax, D. M., & Duin, R. P. (2004). Support vector data description. *Machine Learning*, *54*(1), 45–66.

Tsai, C. C., Wu, T. H., & Lai, S. H. (2022). Multi-scale patch-based representation learning for image anomaly detection and segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 3992–4000).

Valerio Massoli, F., Falchi, F., Kantarci, A., Akti, Ş., Kemal Ekenel, H., & Amato, G. (2020). MOCCA: Multi-layer one-class classification for anomaly detection. arXiv e-prints, arXiv–2012.

Vaze, S., Han, K., Vedaldi, A., & Zisserman, A. (2021). Open-set recognition: A good closed-set classifier is all you need. http://dx.doi.org/10.48550/ARXIV.2110.06207, URL: https://arxiv.org/abs/2110.06207.

Venkatakrishnan, A. R., Kim, S. T., Eisawy, R., Pfister, F., & Navab, N. (2020). Self-supervised out-of-distribution detection in brain CT scans. arXiv preprint arXiv:2011.05428.

Venkataramanan, S., Peng, K. C., Singh, R. V., & Mahalanobis, A. (2020). Attention guided anomaly localization in images. In *European conference on computer vision* (pp. 485–503). Springer.

Villa-Perez, M. E., Alvarez-Carmona, M. A., Loyola-Gonzalez, O., Medina-Perez, M. A., Velazco-Rossell, J. C., & Choo, K. K. R. (2021). Semi-supervised anomaly detection algorithms: A comparative summary and future research directions. *Knowledge-Based Systems*, *218*, Article 106878. http://dx.doi.org/10.1016/j.knosys.2021.106878, URL: https://www.sciencedirect.com/science/article/pii/S0950705121001416.

Vondrick, C., Shrivastava, A., Fathi, A., Guadarrama, S., & Murphy, K. (2018). Tracking emerges by colorizing videos. In *Proceedings of the European conference on computer vision* (pp. 391–408).

Wang, H., Bah, M. J., & Hammad, M. (2019). Progress in outlier detection techniques: A survey. *IEEE Access*, *7*, 107964–108000.

Wang, C., Dou, Y., Chen, M., Chen, J., Liu, Z., & Philip, S. Y. (2021). Deep fraud detection on non-attributed graph. In *2021 IEEE international conference on big data* (pp. 5470–5473). IEEE.

Wang, X., & Gupta, A. (2015). Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision* (pp. 2794–2802).

Wang, G., Han, S., Ding, E., & Huang, D. (2021). Student-teacher feature pyramid matching for unsupervised anomaly detection. arXiv preprint arXiv:2103.04257.

Wang, R., Liu, C., Mou, X., Gao, K., Guo, X., Liu, P., et al. (2023). Deep contrastive one-class time series anomaly detection. arXiv:2207.01472.

Wang, Y., Qin, C., Wei, R., Xu, Y., Bai, Y., & Fu, Y. (2021). SLA^{2}P: Self-supervised anomaly detection with adversarial perturbation. http://dx.doi.org/10.48550/ARXIV.2111.12896, URL: https://arxiv.org/abs/2111.12896.

Wei, D., Lim, J. J., Zisserman, A., & Freeman, W. T. (2018). Learning and using the arrow of time. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8052–8060).

Weng, L., & Kim, J. W. (2021). Tutorial: Self-supervised learning. In A. Canziani, & E. Grant (Eds.), *Advances in neural information processing systems*. URL: https://nips.cc/virtual/2021/tutorial/21895.

Winkens, J., Bunel, R., Roy, A. G., Stanforth, R., Natarajan, V., Ledsam, J. R., et al. (2020). Contrastive training for improved out-of-distribution detection. http://dx.doi.org/10.48550/ARXIV.2007.05566, URL: https://arxiv.org/abs/2007.05566.

Wu, Z., Xiong, Y., Yu, S. X., & Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Xia, X., Pan, X., Li, N., He, X., Ma, L., Zhang, X., et al. (2022). GAN-based anomaly detection: A review. *Neurocomputing*, http://dx.doi.org/10.1016/j.neucom.2021.12.093, URL: https://www.sciencedirect.com/science/article/pii/S0925231221019482.

Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., et al. (2018). Machine learning and deep learning methods for cybersecurity. *IEEE Access*, *6*, 35365–35381.

Xu, Z., Huang, X., Zhao, Y., Dong, Y., & Li, J. (2022). Contrastive attributed network anomaly detection with data augmentation. In *Advances in knowledge discovery and data mining: 26th Pacific-Asia conference, PAKDD 2022, Chengdu, China, May 16–19, 2022, proceedings, part II* (pp. 444–457). Springer.

Xu, J., Zheng, Y., Mao, Y., Wang, R., & Zheng, W. S. (2020). Anomaly detection on electroencephalography with self-supervised learning. In *2020 IEEE international conference on bioinformatics and biomedicine* (pp. 363–368). IEEE.

Yi, J., & Yoon, S. (2020). Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian conference on computer vision*.

Zavrtanik, V., Kristan, M., & Skočaj, D. (2021). DRAEM-A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 8330–8339).

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In M. Meila, & T. Zhang (Eds.), *Proceedings of machine learning research*: *vol. 139*, *Proceedings of the 38th international conference on machine learning* (pp. 12310–12320). PMLR, URL: https://proceedings.mlr.press/v139/zbontar21a.html.

Zeng, X. M., Song, Y., Zhuo, Z., Zhou, Y., Li, Y. H., Xue, H., et al. (2023). Joint generative-contrastive representation learning for anomalous sound detection. In *ICASSP 2023 - 2023 IEEE international conference on acoustics, speech and signal processing* (pp. 1–5). http://dx.doi.org/10.1109/ICASSP49357.2023.10095568.

Zhang, X., Mu, J., Zhang, X., Liu, H., Zong, L., & Li, Y. (2022). Deep anomaly detection with self-supervised learning and adversarial training. *Pattern Recognition*, *121*, Article 108234. http://dx.doi.org/10.1016/j.patcog.2021.108234, URL: https://www.sciencedirect.com/science/article/pii/S0031320321004155.

Zhang, J., Saleeby, K., Feldhausen, T., Bi, S., Plotkowski, A., & Womble, D. (2021). Self-supervised anomaly detection via neural autoregressive flows with active learning. In *NeurIPS 2021 workshop on deep generative models and downstream applications*. URL: https://openreview.net/forum?id=LdWEo5mri6.

Zhang, X., Xie, W., Huang, C., Zhang, Y., & Wang, Y. (2021). Self-supervised tumor segmentation through layer decomposition. arXiv preprint arXiv:2109.03230.

Zhang, Z., Zhao, L., Cai, D., Feng, S., Miao, J., Guan, Y., et al. (2022). Time series anomaly detection for smart grids via multiple self-supervised tasks learning. In *2022 IEEE international conference on knowledge graph* (pp. 392–397). Los Alamitos, CA, USA: IEEE Computer Society, http://dx.doi.org/10.1109/ICKG55886.2022.00057, URL: https://doi.ieeecomputersociety.org/10.1109/ICKG55886.2022.00057.

Zhao, H., Li, Y., He, N., Ma, K., Fang, L., Li, H., et al. (2021). Anomaly detection for medical images using self-supervised and translation-consistent features. *IEEE Transactions on Medical Imaging*, *40*(12), 3641–3651.

Zheng, Y., Jin, M., Liu, Y., Chi, L., Phan, K. T., & Chen, Y. P. P. (2021). Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*.

Zheng, Y., Jin, M., Liu, Y., Chi, L., Phan, K. T., Pan, S., et al. (2022). From unsupervised to few-shot graph anomaly detection: A multi-scale contrastive learning approach. arXiv preprint arXiv:2202.05525.